

Applied Statistics Qualifying Exams Coaching

Michael Howes*

Summer 2023

Contents

0	Applied 2011: Solution	2
1	Applied 2012: Solution	9
2	Applied 2013: Solution	17
3	Applied 2014: Solution	23
4	Applied 2015: Solution	36
5	Applied 2016: Solution	43
6	Applied 2017: Solution	57
7	Applied 2018: Solution	68
8	Applied 2019: Solution	79
9	Applied 2020: Solution	93
10	Applied 2021: Solution	104
11	Applied 2022	115

*With lots of content credit given to previous applied quals coaches

0 Applied 2011: Solution ¹

Problem 1: Multiple-testing across genes

Key ideas/tools:

- recognizing multiple testing
- Bonferroni correction

With the Bonferroni correction, 0.0003 becomes 0.3, so this p -value is not so impressive. On the other hand, IPF1 is not subject to the same multiple comparisons issue since it was pre-selected before looking at the data. Hence, there's no need to "abandon" IPF1. To try and lend support to IPF1, Dr. Fatti should try to present evidence that he didn't p-hack and was genuinely only looking at the association between IPF1 and insulin production.

Other observations: Note that it is possible for multiple genes to affect insulin production, so there isn't necessarily a binary choice between IPF1 and RG45. It is also possible that many genes show up as significant in this setting because we are testing for marginal association, so genes that are correlated with each other can all show up as significant. Assuming that in John Newsense's analysis, RG45 had the lowest p -value, we are not facing the latter issue of finding genes that aren't directly associated with insulin due to correlations between genes.

Problem 2: Poisson GLM modeling of call centers

Key ideas/tools:

- Poisson loglinear models
- using L_1 -penalization to promote sparsity

First of all, the problem seems to suggest we actually know the rates α_i , but that would trivialize things, so let's assume α_i are unknown and are to be estimated through historical data.

- (a) A simple model for a day's worth of historical data is $H_i \stackrel{\text{ind}}{\sim} \text{Poi}(\alpha_i)$, and after perturbation by β_i , we have $X_i \stackrel{\text{ind}}{\sim} \text{Poi}(\alpha_i \cdot \beta_i)$. Note that independence might be a bad or not bad assumption depending on the application. Often "traffic" (e.g. at call centers) is modeled using a Poisson process, which has independent increments. However, if it is phone traffic just between two phone numbers, then independence doesn't seem reasonable.

¹Will Fithian, Gene Katsevich, Kenneth Tay, Stephen Bates, Nikos Ignatiadis, Dan Kluger and M.H.

- (b) If we try to fit separate parameters for every single time interval, then the maximum likelihood estimates are $\hat{\alpha}_i = H_i$, $\hat{\beta}_i = X_i/H_i$. Depending on how much traffic there actually is, these might or might not be reasonable estimates. For example, if there's not much traffic, and $H_i = 0$ for some i , then $\hat{\beta}_i$ is not even defined. On the other hand, if there is a lot of traffic then the standard deviation of $\hat{\alpha}_i$ is $\sqrt{\alpha_i^*}$ which is small relative to α_i^* when α_i^* is large. Here we use α_i^* to denote the true value of the parameters in the case where the model is true.

An alternative approach is to fit a smooth model to the traffic rate:

$$\log \alpha_i = \sum_{j=1}^k \gamma_j h_j(t_i); \quad \log(\alpha_i \cdot \beta_i) = \sum_{j=1}^k \delta_j h_j(t_i),$$

where t_i is the time at the midpoint of interval i , and h_j are a set of cubic spline functions. Then, we can fit Poisson regressions to H_i and X_i to obtain $\hat{\gamma}_j$ and $\hat{\delta}_j$, and then define

$$\hat{\alpha}_i = \exp \left(\sum_{j=1}^k \hat{\gamma}_j h_j(t_i) \right); \quad \hat{\beta}_i = \exp \left(\sum_{j=1}^k (\hat{\delta}_j - \hat{\gamma}_j) h_j(t_i) \right).$$

- (c) Instead of modeling β_i smoothly using a basis expansion, let's fit each β_i separately, but use a lasso penalty to regularize. Let $b_i = \log(\beta_i)$. Then, we assume

$$\log \alpha_i = \sum_{j=1}^k \gamma_j h_j(t_i); \quad \log(\alpha_i \cdot \beta_i) = b_i + \sum_{j=1}^k \gamma_j h_j(t_i).$$

We can throw this all into one big Poisson regression and then regularize using an L_1 penalty on b_i .

- (d) If we had (H^1, \dots, H^M) , then $\sum_{m=1}^M H_i^m \sim \text{Poi}(M\alpha_i)$ would be sufficient, so this essentially reduces to the previous problem. Here are two reasons why this could be useful: Perhaps, if we really have many days of traffic data (and also expect that traffic times have not changed over these days), then perhaps we would have enough data to abandon the smooth basis expansion for the modeling of the historical $\log(\alpha_i)$ and include each α_i as an individual parameter in the model (the other considerations described above would still apply). Furthermore, having several days of historical data may be useful for a different reason: it affords us a simple way to check the Poisson assumption. e.g. we could check for over-dispersion and if we detect over-dispersion, we can fit a negative binomial GLM instead.

Problem 3: Pearson χ^2 puzzle

Key ideas/tools:

- definition of χ^2 test

Suppose we have the following expected counts:

E_{11}	E_{12}
E_{21}	E_{22}

Under the assumption of independence, the expected counts become:

$$\begin{aligned} E_{11} &= \frac{(a+b)(a+c)}{a+b+c+d} & E_{12} &= \frac{(a+b)(b+d)}{a+b+c+d} \\ E_{21} &= \frac{(c+d)(a+c)}{a+b+c+d} & E_{22} &= \frac{(c+d)(b+d)}{a+b+c+d}. \end{aligned}$$

Recall that in this case the pearson χ^2 statistic is

$$X^2 := \frac{(a - E_{11})^2}{E_{11}} + \frac{(b - E_{12})^2}{E_{12}} + \frac{(c - E_{21})^2}{E_{21}} + \frac{(d - E_{22})^2}{E_{22}}, \quad (1)$$

and this has a limiting χ^2 distribution with 1 degree of freedom. As a result, a test will be rejected at the .05 level if $X^2 \geq 1.96^2 \approx 4$.

Intuitively, we have two ways of making X^2 large. One option is to set $a = 1$ and take $b = c$ large (that is $(0, 0)$ occurs much less frequently than $(0, 1)$ and $(1, 0)$). The other option is to take $b = c = 1$ and a large (that is $(0, 0)$ occurs much more frequently than $(0, 1)$ and $(1, 0)$). It turns out the second option will give us what we want. This is because the question requires $d \geq 1$.

We will show that if $a \gg b, c$, then the chi-squared test will reject as long as $d \geq 1$. Set $b = c = 1$, then $E_{22} = \frac{(d+1)^2}{a+d+2}$, and the last term in (1) becomes

$$\frac{(d - E_{22})^2}{E_{22}} = E_{22} - 2d + \frac{d^2}{E_{22}} \geq -2d + \frac{d^2}{(d+1)^2}(a+d+2) \geq -2d + \frac{1}{4}(a+d+2),$$

where the last inequality uses the fact that $d \geq 1$. So, for example, if we set $a = 10^6$, the χ_1^2 statistic is extremely significant for $1 \leq d \leq 10^5$ due to this term. On the other hand, if $a, d > 10^5$ then $E_{12} = \frac{(a+1)(d+1)}{a+d+2} \gg 1$ and the χ_1^2 statistic is extremely significant due to the b term in (1). Therefore with $a = 10^6, b = c = 1$, and any $d \geq 1$, the Pearson χ^2 test of independence will be rejected at the 0.05 level.

Problem 4: Assessing time series correlations

Key ideas/tools:

- time series have correlated data points
- block bootstrap or randomization for timeseries

The problem doesn't seem to be going for this, but one thing to note is that the wily graduate student has done “a bit of searching in a published table of stock market trends” to find SKRN. Even if the test for correlation was valid, it would need to be corrected for how many other stocks the graduate student looked at.

- (a) One way to test for correlation is to test significance in a linear regression of say $Y_i \sim X_i$. The p-value produced by `cor.test` is made under the assumption that (X_i, Y_i) i.i.d. normal with correlation 0. Under this assumption,

$$\frac{\sqrt{n-2}r_n}{\sqrt{1-r_n^2}} \sim t_{n-2},$$

where r_n is the sample correlation, n is the number of samples and t_{n-2} is the t distribution with $n-2$ degrees of freedom.

For the confidence interval `cor.test` uses Fisher's transformation: recall

$$\operatorname{arctanh}(u) = 1/2 \log((1+u)/(1-u)).$$

If all the pairs (X_i, Y_i) are i.i.d. from some joint distribution with correlation ρ , then

$$\operatorname{arctanh}(r_n) \approx \mathcal{N}\left(\operatorname{arctanh}(\rho), \frac{1}{n-3}\right).$$

Thus, both the p-value and the confidence interval are made under the assumption that the pairs (X_i, Y_i) are i.i.d.

In any case, we are making an assumption that the different time points are independent, which is contradicted by a cursory glance at the data. The non-independence will lead our t statistic to have much higher variance than its nominal null distribution.

- (b) If the pairs (X_i, Y_i) were really i.i.d. pairs, we could do a standard permutation test (i.e. permute the Y_i and leave the X_i fixed) to obtain null distributions of whatever test statistic we might like. Alternatively, we could bootstrap by resampling (X_i, Y_i) pairs with replacement. Both these methods will fail due to the temporal dependence.
- (c) We could use the block bootstrap to create a confidence interval for the correlation coefficient, and check if the block bootstrap confidence interval contains 0. Looking at the data, it seems reasonable to choose 20 blocks (each with 10 consecutive days) or to choose 10 blocks (each with 20 consecutive days).

Problem 5: Entropy of Gaussians, (sparse) PCA and “ICA”

Key ideas/tools:

- directly computing the entropy
- definition of PCA
- Ideas from ICA (FastICA) and sparse PCA?

(a) Note that $Z = \Sigma^{-1/2}(X - \mu)$ has standard normal distribution. Thus,

$$\begin{aligned} H(X) &= -\mathbb{E} \left[\frac{-1}{2} (X - \mu)^\top \Sigma^{-1} (X - \mu) - \frac{1}{2} \log |\Sigma| - \frac{p}{2} \log 2\pi \right] \\ &= \frac{1}{2} \mathbb{E} [Z^\top Z] + \frac{1}{2} \log |\Sigma| + \frac{p}{2} \log 2\pi \\ &= \frac{1}{2} (p + \log |\Sigma| + p \log 2\pi). \end{aligned}$$

(b) $Y = a^\top X \sim N_1(a^\top \mu, a^\top \Sigma a)$, so by part (a), its entropy is

$$H(Y) = \frac{1}{2} (1 + \log a^\top \Sigma a + \log 2\pi).$$

Maximizing $H(Y)$ is equivalent to maximizing $a^\top \Sigma a$. Thus, the entropy-maximizing direction is the first eigenvector of Σ .

(c) The MLE for an IID sample of multivariate Gaussians is

$$\hat{\mu} = \bar{x}_N = \frac{1}{N} \sum_{i=1}^N x_i \quad \text{and} \quad \hat{\Sigma} = \frac{1}{N} \sum_{i=1}^N (x_i - \bar{x}_N)(x_i - \bar{x}_N)^\top.$$

This fact can be derived by hand or see equation (4.2.3) in Mardia et al. [1979], among others, for a reference about this fact. \hat{a} is the unit vector in the direction of the leading eigenvector of $\hat{\Sigma}$, which by definition of PCA is the first principal component direction of the data.

(d) It's not too clear what the question is going for, perhaps one interpretation is that we seek to find the first direction from a sparse PCA approach. That is to solve

$$\begin{aligned} &\text{maximize} && a^\top \hat{\Sigma} a - \lambda \|a\|_1 \\ &\text{subject to} && \|a\|_2 = 1 \end{aligned}$$

This problem is not convex, but it can be solved heuristically. As for not assuming a parametric form of the density, one can give non-parametric justifications for PCA, and so the above is still a reasonable answer.

Another interpretation is that we seek to find a direction that maximizes the population entropy. Let $\sigma^2(a) = a^\top \Sigma a$, the variance along the a -th direction.

For simplicity also assume that $\mu = 0$. Also assume that P is absolutely continuous w.r.t. the Lebesgue measure. Then let \tilde{p}_a be the Normal distribution with variance $\sigma^2(a)$ and note

$$\begin{aligned}\mathbb{E}[-\log(P_a)] &= - \int \log(p_a) p_a d\lambda \\ &= - \int \log(p_a/\tilde{p}_a) p_a d\lambda - \int \log(\tilde{p}_a) p_a d\lambda \\ &= -D_{\text{KL}}(p_a, \tilde{p}_a) + \frac{1}{2}(1 + \log(\sigma^2(a)) + \log 2\pi).\end{aligned}$$

So we want to find a direction that leads to both a large variance and is close in KL-Divergence to a Normal distribution with the same variance (note KL divergence is affine invariant too). One way to approach this would be to try similar heuristics as in FastICA. Sparsity could be enforced by an $L1$ penalty or perhaps a forward-stepwise approach.

Problem 6: Logistic regression modeling with spatial data

Key ideas/tools:

- correlations within spatial data
- cross-validation with correlated data points

(a) (i) We model

$$Y_i \stackrel{\text{ind}}{\sim} \text{Ber}(p_i), \quad \text{logit}(p_i) = \beta_0 + \beta^\top X_i. \quad (2)$$

Once we fit $\hat{\beta}_0, \hat{\beta}$, we classify a new observation X^* based on $Y^* = I(\hat{\beta}_0 + \hat{\beta}^\top X^* > 0)$. Hence, this classifier has a linear decision boundary.

- (ii) We can estimate the classification error using either cross-validation or a held-out test set.
- (iii) Equation (2) states our assumptions, namely that the Y_i are independent given X_i , and follow the Bernoulli distribution with the given parametric form.

(b) **Assessing classification accuracy**

First note that if the main goal is classification, then it is not very crucial to check all the statistical assumptions of logistic regression. Since the scientist is primarily interested in classification accuracy, the main thing we should be worried about is that our cross-validation estimate of accuracy is not valid when the data points are not independent. In this case, we are probably concerned primarily about some sort of spatial or temporal correlation among the samples. We can check for spatial dependence by plotting a heatmap of the residuals (see

below), with coordinates given by the coordinates of each sampled region. A non-random pattern here would suggest spatial correlation. We can do the same for temporal correlation if the samples were collected at different times.

If there is important spatial structure, we can instead do a grouped cross-validation, where we put points that are spatially close into the same fold. This will ensure that the test points are nearly independent of the training points for each fold, which will result in a more reliable estimate of classification accuracy.

Assessing linearity and independence

Modeling assumptions matter most when we want to do inference based on the model. If the environmental scientist still cares about model fit, then we could do the following.

- (i) Diagnosing both linearity and independence are usually done by looking at various plots involving residuals. For logistic regression, we can define Pearson residuals via

$$r_i = \frac{Y_i - \hat{p}_i}{\sqrt{\hat{p}_i(1 - \hat{p}_i)}}.$$

To check linearity, we can plot these residuals versus each of the predictor variables. Seeing a non-random pattern might suggest that high-order terms or interactions need to be included in the model. (We can also look at deviance residuals, note though that in general “residuals” for general GLMs are hard to interpret compared to linear regression residuals.)

- (ii) We can augment the logistic regression with higher order terms or interactions if the linearity of the model is in question, though once we start adding lots of these terms we should start regularizing (see also part (c)). If there is spatial or temporal autocorrelation, we could add extra spatial or temporal spline terms to the regression to mitigate this effect. If we care about finding standard errors for our coefficients in the presence of autocorrelation, we could bootstrap by resampling spatial blocks of data points.
- (c) Some form of dimension reduction or regularization would be advisable in this context. As far as dimension reduction goes, principal components regression would apply if we had correlated features. An alternative is to regularize. In particular, if we think a small subset of the predictors is relevant, we can apply L_1 regularization; if predictors are correlated we can apply L_2 regularization; if both apply then we could use elastic net regularization. To choose the tuning parameters, we should perform cross-validation using folds that respect the spatial structure, as described in part (b).

1 Applied 2012: Solution ²

Problem 1: Deviance manipulations

Key ideas/tools:

- deviance calculations
- The MLE in exponential families matches the observed means of the sufficient statistics

(a) This is a straight-forward manipulation:

$$\begin{aligned}\log \frac{f_{\alpha}(\hat{\beta})}{f_{\alpha_0}(\hat{\beta})} &= \alpha' \hat{\beta} - \psi(\alpha) - [\alpha_0' \hat{\beta} - \psi(\alpha_0)] \\ &= (\alpha - \alpha_0)'(\hat{\beta} - \beta_0) + (\alpha - \alpha_0)' \beta_0 - [\psi(\alpha) - \psi(\alpha_0)] \\ &= Q(\hat{\beta}) - D(\alpha_0, \alpha)/2,\end{aligned}$$

as required.

(b) The notation here can be somewhat confusing. It is useful to recall that $\beta_0 := \mathbb{E}_{\alpha_0}[\hat{\beta}]$. Then, Hoeffding's formula says that

$$\log \frac{f_{\alpha}(\hat{\beta})}{f_{\hat{\alpha}}(\hat{\beta})} = -D(\hat{\alpha}, \alpha)/2,$$

where $\hat{\alpha}$ is the MLE. Now under the MLE we have that $\beta_0 = \mathbb{E}_{\alpha_0=\hat{\alpha}}[\hat{\beta}] = \hat{\beta}_{obs}$ where $\hat{\beta}_{obs}$ denotes the observed data. Thus, Hoeffding's formula is the same as our previous result specialized to $\alpha_0 = \hat{\alpha}$, since in this case $Q = 0$ because $\beta_0 = \hat{\beta}$, i.e. the fitted mean under the MLE is equal to the observed data.

(c) By part (a),

$$\mathbb{E}_{\alpha_0} \left[e^{Q(\hat{\beta})} \right] = \mathbb{E}_{\alpha_0} \left[\frac{f_{\alpha}(\hat{\beta})}{f_{\alpha_0}(\hat{\beta})} e^{D(\alpha_0, \alpha)/2} \right] = \mathbb{E}_{\alpha_0} \left[\frac{f_{\alpha}(\hat{\beta})}{f_{\alpha_0}(\hat{\beta})} \right] e^{D(\alpha_0, \alpha)/2} = e^{D(\alpha_0, \alpha)/2}.$$

Above we used the following

$$\mathbb{E}_{\alpha_0} \left[\frac{f_{\alpha}(\hat{\beta})}{f_{\alpha_0}(\hat{\beta})} \right] = \int \frac{f_{\alpha}(\hat{\beta})}{f_{\alpha_0}(\hat{\beta})} f_{\alpha_0}(\hat{\beta}) d\beta = \int f_{\alpha}(\hat{\beta}) = 1,$$

since f_{α} is the density for $\hat{\beta}$ at the parameters α .

²Will Fithian, Gene Katsevich, Kenneth Tay, Stephen Bates, Nikos Ignatiadis, Isaac Gibbs, Dan Kluger, M.H.

Problem 2: Weighted least squares and optimal design

Key ideas/tools:

- weighted least squares

(a) First, we transform the problem to the homoskedastic linear regression model:

$$x^{-r}y = x^{-r+1}\beta + \varepsilon; \quad \varepsilon \sim \mathcal{N}(0, \sigma^2 I_n). \quad (3)$$

The OLS estimate in this model is

$$\hat{\beta} = \frac{\sum_{i=1}^n x_i^{-2r+1} y_i}{\sum_{i=1}^n x_i^{-2r+2}}.$$

Equivalently, $\hat{\beta}$ is the weighted least squares estimator using the diagonal covariance matrix $\Sigma = \sigma^2 \text{diag}(x^{2r})$.

(b) For this part we view the x values as fixed. From (3), we see that $\text{Var}(x^{-r}y) = \sigma^2$. Hence,

$$\begin{aligned} \text{Var}[\hat{\beta}] &= \frac{1}{\left(\sum_{i=1}^n x_i^{-2r+2}\right)^2} \sum_{i=1}^n \text{Var}(x_i^{-2r+1} y_i) \\ &= \frac{1}{\left(\sum_{i=1}^n x_i^{-2r+2}\right)^2} \sum_{i=1}^n x_i^{-2r+2} \text{Var}(x_i^{-r} y_i) \\ &= \frac{1}{\left(\sum_{i=1}^n x_i^{-2r+2}\right)^2} \sum_{i=1}^n x_i^{-2r+2} \sigma^2 \\ &= \frac{\sigma^2}{\sum_{i=1}^n x_i^{-2r+2}}. \end{aligned}$$

(c) If $r = 1/2$, then the variance of our estimate is inversely proportional to $\sum_{i=1}^n x_i$. Hence, we seek to maximize $\sum_{i=1}^n x_i$ subject to the constraint that $n + \sum_{i=1}^n x_i \leq 100$. It is clear that the optimal solution to this is $n = 1$ and $x_1 = 99$. Note that if we were sure of our linear model, then this is the most efficient choice. However, usually we are not as confident that our model is correct, so fitting a linear regression using only one data point would be a bad idea because we could not use that data point to check the model fit.

Problem 3: Multiple testing with gene expression data

Key ideas/tools:

- multiple testing

- computations under the global null

- (a) Let us first assume that each p-value is uniform, i.e., $P_i \sim U[0, 1]$ under the null. Then each null hypothesis has probability 0.01 of being rejected, so using the linearity of expectation we expect $0.01 \cdot 7184 \approx 72$ nominally significant genes.

If we have conservative p-values, i.e.,

$$\mathbb{P}_{H_0} [P_i \leq t] \leq t \text{ for all } t, \quad (4)$$

then the above is still an upper bound.

- (b) Let X be the number of nominally significant genes at the 0.01 level. Let's make a big assumption: that expressions of different genes are independent. Under this assumption, we have $X \sim \text{Bin}(7184, 0.01)$ under the null. To get a p-value for $X = 94$, we can use the normal approximation $X \dot{\sim} N(72, 72)$, which leads to $Z = (94 - 72)/\sqrt{72} \approx 2.6$. This leads to a p-value of about 0.005. Alternatively, since the null distribution is known you could also compute an exact p-value.

Note that again this (global null) p-value is a valid p-value, as long as the per-gene p-values are conservative.

If we did not want to make the independence assumption (since in reality gene expressions are in fact correlated), then we could apply a permutation test, repeatedly scrambling the treatment labels of the 100 subjects and recomputing X^* . This would give a finite sample exact test of the null hypothesis that gene expression levels have the same distribution in treated and untreated subjects. Intuitively, it should also give a reasonable asymptotic test of the intersection null hypothesis that all genes have the same mean expression level, but proving this formally would be technically challenging.

- (c) Let X_b be the number of p-values falling into $[0, 0.01]$ and let X_c be the number of p-values falling into $[0, 0.001]$.

Ignoring the information from part (b), we can simply test the intersection null using the same approach from part (b). In particular $X_c \sim \text{Bin}(7184, 0.001)$, so we can use an exact Binomial test, although this is hard to do on an exam without access to a computer. We can instead use a test based normal approximation with a Z -score of $Z_c = (18 - 7.184)/\sqrt{7.184 \times 0.999} \approx 4.04$, and get an approximate p-value of 2.7×10^{-5} . However, it should be noted that the normal approximation for a binomial doesn't work quite as well when the success probability of a trial is so close to 0 or 1 (in fact in this case the exact p-value based on binomial testing is 0.00048, nearly 20 times higher than the normal-based estimate).

There are a few approaches we could use to test the null using information from part (b) and (c).

The first approach is based on a chi-squared test. We have $X_c = 18$ p-values falling into $[0, 0.001]$ and $X_2 = 76$ p-values falling into $[0.001, 0.01]$. Assuming independence and uniformity of the underlying p-values, we have approximately $X_c \sim N(7.2, 7.2)$ and $X_2 \sim N(65, 65)$. Now,

$$\begin{aligned} \text{Cov}(X_c, X_2) &= \sum_{i,j} \text{Cov}(\mathbb{1}_{P_i \leq 0.001}, \mathbb{1}_{0.001 \leq P_j \leq 0.01}) \\ &= \sum_i -0.001 * (0.01 - 0.001) \cong -0.00001 * 7184 \cong -0.072. \end{aligned}$$

This is very small relative to the variances of X_c and X_2 so we can obtain an approximately valid test by treating X_c and X_2 as independent. By doing this we get that $Z_c = (18 - 7.184)/\sqrt{7.184 \times 0.999} \approx 4.04$ and $Z_2 = (76 - 65)/\sqrt{65} \approx 1.4$, and can combine these using the chi squared test ($Z_c^2 + Z_2^2 \sim \chi_2^2$), which results in a p-value of about 0.0001. This is higher than the approximate p-value of 2.7×10^{-5} that ignores part (b), but lower than the actual p-value of 0.00048 when ignoring part (b). Therefore for this approach, whether including the information from part (b) increases or decreases the p-value is very sensitive to the normal approximation being used here and the approximate independence of X_c and X_2 . If one has a computer they should instead find the null distribution of $Z_c^2 + Z_2^2$ by simulating new p-values under the global null to get a finite sample exact test (assuming all p-values are independent).

A major drawback of the first approach (besides not being able to tell whether the information from part (b) increases or decreases the p-value) is that it is essentially a two-sided test in the sense that it will reject the null the p-values are conservative and X_c and X_2 are both very small.

A second approach is the following: Let $t = 0.001$ and $\tau = 0.1$, and suppose that:

$$\mathbb{P}_{H_0} [P_i \leq t \mid P_i \leq \tau] \leq t/\tau, \quad (5)$$

in fact \leq is an equality under the uniformity assumption.

This suggests a second approach: under the global null

$$X_c \mid X_b \sim \text{Bin}(X_b, 0.1),$$

which yields a p-value of about 0.001.

Finally, we note that the 2nd approach of keeping only p-values $\leq \tau$ and using (5) can be useful for multiple testing when null p-values are very conservative (i.e., very far from uniform), see Zhao, Small, and Su [2019] (the first test will reject for small X_b and X_c). Such approaches however do not work under (4), instead (5) is required. Zhao, Small, and Su [2019] call p-values that satisfy (5) for all $t \leq \tau$ “uniformly conservative”.

A third approach would be to the p-values uniformly conservative and independent and compute an upper bound on $\mathbb{P}_{H_0}(X_c \geq 18, X_b \geq 94)$, but this will involve summing over many binomial tail probabilities and cannot easily be done by hand for an approximate p-value.

The same caveats about assuming independence across gene expressions apply for this part as well.

Problem 4: Separation in logistic regression

Key ideas/tools:

- separating hyperplanes in classification
 - regularization
- (a) We have probably found a separating hyperplane (a hyperplane perfectly separating the two classes). This causes the likelihood to diverge, resulting in unstable parameter estimates (note that there are multiple separating hyperplanes, and they are equally good from the point of view of the logistic loss function). We can plot y against $X\hat{\beta}$ to see whether this is indeed the case. This is a problem with the model, not with the data. The data might reflect that these ten predictors can actually separate the classes very well, which would be great! Or we might have overfit. To know if we have overfit or not, we could estimate test error by cross-validation or a held-out test set.
- (b) The OLS model will not fit the data more closely. We will probably see $\hat{\beta}$ oriented in roughly the same direction, but with some negative predictions and some greater than 1. It would not be reasonable to treat these as probability estimates.
- (c) From part (a), it is not clear that we got a bad classifier the first time, only a bad estimate for β and for the conditional probabilities. However, the substantially higher test error indicates that the model could have been overfit. To address the overfitting, we can modify the model by adding some regularization to the parameters (e.g., Ridge, Lasso, max margin).

Comment: In a logistic model with design matrix X and fitted values $\hat{\pi}$, the estimated covariance matrix for $\hat{\beta}$ is

$$\hat{\Sigma} = (X^\top \text{diag}(\hat{\pi}(1 - \hat{\pi}))X)^{-1}.$$

Thus, if $\hat{\pi} \approx 0, 1$, then $\hat{\pi}(1 - \hat{\pi}) \approx 0$ and the $\hat{\Sigma}$ will have large entries. This is why we are getting large fitted values of $\hat{\beta}$ and large standard errors in (a).

In the OLS model the covariance matrix for $\hat{\beta}$ is $(X^\top X)^{-1}$. This matrix should have smaller values than $\hat{\Sigma}$. In part (b), we could look at the covariance matrix from the OLS fit. If this matrix also has large standard errors, then there is probably a

problem with co-linearity in our features. We could talk to the scientist about what the 10 features are and possibly remove some redundant features. This would help reduce the overfitting.

Problem 5: Dimensionality reduction for images

Key ideas/tools:

- PCA and sparse PCA
- (a) Principal component analysis (PCA) would be the natural choice.
- (b) There are various sparse PCA methods we can use. Here is one example due to Witten, Tibshirani, and Hastie [2009]: to obtain the first direction, we solve

$$(u, v, d) = \arg \min \|X - d u v^\top\|_F^2$$

$$\text{subject to } \|u\|_2 = \|v\|_2 = 1, \|u\|_1 \leq c_1, \|v\|_1 \leq c_2,$$

where d is a scalar and u and v are vectors. This is equivalent to solving

$$(u, v) = \arg \max u^\top X v$$

$$\text{subject to } \|u\|_2, \|v\|_2 \leq 1, \|u\|_1 \leq c_1, \|v\|_1 \leq c_2,$$

which is bi-convex in u and v , so we can use some alternating method to solve it. Once we have u and v it is then easy to solve for d .

We can then obtain $X^{(2)} = X - d u v^\top$ and solve the same problem for $X^{(2)}$ to get the next direction. By iterating, we can get multiple directions.

- (c) We can solve the same problem for (say) 4×4 pixel chunks, then replace each pixel chunk with a few principal components and apply the approach from part (b) to the resulting matrix.
- (d) Cross-validation. For each fold, redo all the feature extraction steps with the given choices of tuning parameters.

If we have many tuning parameters, CV might be inappropriate because we are doing a high-dimensional search over tuning parameters. A better approach might be to try to set most of the parameters in an unsupervised fashion, and then only cross-validate to select one or two tuning parameters at the end.

Problem 6: Testing for a discrete distribution

Key ideas/tools

- Bare-hands calculations under the null hypothesis.

- Geometric distribution.

It should be noted that Alice and Bob's statements are not diametrically opposed to each other. For example, it is possible that Q appears in all letters, but with lower frequency than other letters: in this setting both Alice and Bob are wrong. Also it is possible that Q only appears as the middle letter but appears in 3 out of every 26 licence plates and has the same overall frequency as other letters: in this setting both Alice and Bob are right.

To perform hypothesis testing we will need to define two disjoint hypotheses, a null and an alternative. If Alice's statement is the null hypothesis, then it will be hard to conduct a hypothesis test, because Alice's statement doesn't specify a null distribution. Therefore, we should make the null hypothesis reflect Bob's statement rather than Allison's. Let (X_i, Y_i, Z_i) denote the first, second and third letters of the i th car. A reasonable null hypothesis, which implies that with Bob's statement is true, is that

$$H_0 : (X_i, Y_i, Z_i) \stackrel{IID}{\sim} \text{Unif}(\{A, B, \dots, Y, Z\}^3).$$

The alternative hypothesis can be Alice's statement. In particular our alternative is

$$H_A : \mathbb{P}(X_i = Q \text{ or } Z_i = Q) = 0 \quad \forall i.$$

- Let N be the number of cars we see before one of the X_i or Z_i is a Q. Then under H_0 , $N \sim \text{Geom}(1 - (25/26)^2)$. We can reject Bob's null hypothesis as soon as N passes its $1 - \alpha$ quantile under the null, whereas as soon as we see a Q in position 1 or 3 we can reject Alice's alternative.
- We need to calculate the $1 - \alpha$ quantile of N under H_0 . $\mathbb{P}(N > n) = (25/26)^{2n}$. So we set $\alpha = (25/26)^{2n}$ and obtain $n = \lceil \frac{1}{2} \log_{25/26}(\alpha) \rceil$.
- Alternatively, we could count the number of "Q"s that we observe in the middle slot before we observe a "Q" in the first or third slot. Let N be the number of "Q"s that we observe in the middle slot before observing any "Q"s in the first or third slot. Then under the null, $N \sim \text{Geom}(2/3)$. The critical threshold is then the $1 - \alpha$ quantile of this distribution, which we will call N_0 . If we see $N \geq N_0$ cars with a Q in the center slot before seeing any with a Q in the first or third slot, we would cease collecting data and reject the null hypothesis. If we observe any "Q"s in the first or third slot before then we could reject Alice's claim.

The rejection threshold when $\alpha = .05$ is the smallest value N_0 such that $P(N \geq N_0) = (1/3)^{N_0} < .05$, which is $\log(.05)/\log(1/3) \approx 2.73$. Thus, if we observe 3 cars with a "Q" in the center slot, we would reject the null hypothesis. The number of cars that would be needed is random in this case, but under Bob's hypothesis we would need $3 \cdot 26$ cars in expectation because the number of cars with a "Q" in the center slot is distributed as $\text{Geom}(1/26)$

We remark that we could have made different choices for H_0 and H_A . For example, let π_Q denote the probability that any arbitrary license plate letter is a Q. A null that reflects Bob's statement along with a simple alternative that is consistent with Alice's statement is

$$H_0 : \pi_Q = \frac{1}{26} \quad \text{and} \quad H_A : \pi_Q = \frac{1}{3 \times 26}.$$

This can be tested by simply considering the 3n letters (ignoring their order) and using Binomial testing in part (a) and then a power calculation which uses normal approximations to the binomial in part (b), but this approach will be messier and require more calculations (here our results will be in terms of some prespecified level α and power β , whereas in the setup originally considered, the power is 1, so there is no need to consider power in part (b)). This approach also doesn't really relate to Alice's statement that Q can only be in the 2nd position or leverage the order of the letters.

2 Applied 2013: Solution³

Problem 1: Finding the optimal line through a point cloud

Key ideas/tools:

- setting up objective functions
- solving non-convex optimization problems

As the City Statistician, part of your job is to talk to the City Council and clarify exactly what they want. There might be a few different options, each could lead to a valid answer.

1. Perhaps they want the houses to be close to the railroad track on average.
2. Perhaps the City Physicist has a model for how the noise of the siren travels and thus the probability that a resident will hear the siren if her house is at a given distance from the railroad track. The City Council might then want to maximize the sum of the probabilities of hearing the siren across all residents.
3. Or, perhaps the City Council is interested in a worst-case analysis, and wants to minimize the maximum distance from any house to the railroad track.

Let $x_1, \dots, x_n \in \mathbb{R}^2$ be the locations of the houses. We can parameterize the train tracks using a normal vector η of unit length and an offset c , such that $d_{\eta,c}(x_i) = |\eta^\top x_i + c|$. Let's go through the three options.

1. In this case, we might want to minimize the average distance to the tracks:

$$\text{minimize } \frac{1}{n} \sum_{i=1}^n |\eta^\top x_i + c|, \quad \text{subject to } \|\eta\|_2 = 1.$$

Note that this problem is non-convex due to the constraint $\|\eta\|_2 = 1$. Nevertheless, it can be solved using a 2D grid search over (η, c) . In an ideal physical setting, sound follows an inverse square law, and so we could instead minimize the average squared distance

$$\text{minimize } \frac{1}{n} \sum_{i=1}^n (\eta^\top x_i + c)^2, \quad \text{subject to } \|\eta\|_2 = 1.$$

This optimization problem is solved by PCA. The train track will run along the first principal component. Equivalently, the optimal η is orthogonal to the first

³Stefan Wager, Gene Katsevich, Kenneth Tay, Nikos Ignatiadis, M.H.

principal component of the centered locations $z_i = x_i - \bar{x}$ and the optimal c is $-\eta^\top \bar{x}$ (so that the train line will pass through the location \bar{x}).

This connection with PCA is probably why this question was asked. It's a nice connection, but the other answers are valid and may be better for actually creating a warning system.

2. Option 2 might depend on what the physical model is for the probability of hearing the siren, but it'll probably end up being a non-convex optimization problem we need to solve by grid search.

3. Option 3 corresponds to

$$\text{minimize } \max_i |\eta^\top x_i + c|, \quad \text{subject to } \|\eta\|_2 = 1,$$

which again is convex except for the constraint.

Problem 2: Dropout in linear regression

Key ideas/tools:

- writing out the score equation
- analytic formulation of ridge regression

Some context: This problem follows the paper by Wager, Wang, and Liang [2013] who propose a simple model to study dropout [Srivastava et al., 2014], a regularization method used to train deep nets.

Solution: Often it's easier to work with vectors and matrices than with individual elements. Let X_I be the (random) design matrix that is the element-wise product of X and I (i.e., $X_I = X \odot I$, where \odot is the Hadamard product of two matrices). Then, our loss function is

$$L_I = \|y - X_I \beta\|^2.$$

For fixed I , the score equations are

$$0 = \nabla_\beta L_I = -X_I^\top (y - X_I \hat{\beta}) \iff X_I^\top y = X_I^\top X_I \hat{\beta}.$$

Now, let us calculate the expectation of both sides of the above expression with respect to the randomness in I . First of all, note that $\mathbb{E}[I_{ij}] = 1$ and $\mathbb{E}[I_{ij}^2] = 1/\phi$. It follows that $\mathbb{E}[X_I] = X$, and

$$(\mathbb{E}[X_I^\top X_I])_{j_1 j_2} = (X^\top X)_{j_1 j_2}, \quad j_1 \neq j_2$$

and

$$(\mathbb{E}[X_I^\top X_I])_{jj} = \frac{1}{\phi} (X^\top X)_{jj}.$$

Putting it together,

$$\mathbb{E}[X_I^\top X_I] = X^\top X + \frac{1-\phi}{\phi} \text{diag}(X^\top X),$$

where $\text{diag}(X^\top X)$ in this case denotes the diagonal matrix with diagonal coinciding with that of $X^\top X$. Hence, the expected score equations are

$$X^\top y = \left(X^\top X + \frac{1-\phi}{\phi} \text{diag}(X^\top X) \right) \hat{\beta}.$$

Hence, we are adding terms to the diagonal of $X^\top X$ before inverting, which is similar to ridge regression. If the columns of X are all normalized to have unit norm, then $\text{diag}(X^\top X)$ becomes the identity, and the result is exactly ridge regression, with regularization parameter $(1-\phi)/\phi$.

We can view dropout as adding noise to the features, which will perturb the fit more when $\|\beta\|_2$ is large. Thus, both dropout and ridge make you “pay” for using a needlessly large β .

Additional question: What if we repeat the above question with additive noise? Say, if instead of $X_I = X \odot I$ we consider $X_E = X + E$ where $E_{ij} \stackrel{\text{iid}}{\sim} \mathcal{N}(0, \sigma^2)$.

Problem 3: Combining quantile estimators

Key ideas/tools:

- normal approximation of sample quantiles
- combining estimates with weighted least squares

Let $\alpha = 0.05$. Assuming n_i are large enough, we have the normal approximation

$$Q_i \sim N \left(Q^\alpha, \frac{1}{n_i} \frac{\alpha(1-\alpha)}{f^2(Q^\alpha)} \right),$$

where f is the density of Y (van der Vaart Cor 21.5). This density f is unknown. We can write the problem as the following homoskedastic linear regression model:

$$\sqrt{n_i} Q_i = \sqrt{n_i} Q^\alpha + \varepsilon_i, \quad \varepsilon_i \stackrel{\text{iid}}{\sim} N(0, \sigma^2),$$

where $\sigma^2 = \frac{\alpha(1-\alpha)}{f^2(Q^\alpha)}$ and $i = 1, \dots, N$. From this we get the MLE

$$\hat{Q}^\alpha = \frac{\sum_{i=1}^N n_i Q_i}{\sum_{i=1}^N n_i}.$$

Our variance estimate for \hat{Q}^α is

$$\widehat{\text{Var}}[\hat{Q}^\alpha] = \frac{\hat{\sigma}^2}{\sum_{i=1}^N n_i},$$

where

$$\hat{\sigma}^2 = \frac{1}{N-1} \sum_{i=1}^N n_i (Q_i - \hat{Q}^\alpha)^2.$$

Problem 4: Dependence in time series regression

Key ideas/tools:

- correlated data points in linear regression
- block bootstrap

(a) In linear regression, $\hat{\beta} = (X^\top X)^{-1} X^\top Y$, so

$$\text{Var}(\hat{\beta}) = (X^\top X)^{-1} X^\top \Sigma X (X^\top X)^{-1},$$

where $\Sigma = \text{Var}(Y)$. The `lm` function in R assumes that $\Sigma = \sigma^2 I$, i.e., that the noise is homoskedastic and uncorrelated. With that assumption, the variance simplifies to $\text{Var}(\hat{\beta}) = \sigma^2 (X^\top X)^{-1}$. With auto-correlated data, however, Σ is not even close to being a multiple of I , so this approach does not work.

(b) As a simple example, suppose that $X = (1, 1, \dots, 1)^\top$ so our model is,

$$Y_i = \beta + \varepsilon_i,$$

where $\beta \in \mathbb{R}$ is a scalar modelling the mean of Y and $\varepsilon \sim \mathcal{N}(0, \Sigma)$. In this case the OLS estimate for β is $\hat{\beta} = \bar{Y}$ and

$$\text{Var}(\hat{\beta}) = \frac{1}{n^2} \sum_{i,j} \Sigma_{ij}.$$

If $\Sigma = \sigma^2 I$, this reduces to σ^2/n . On the other hand, suppose Σ is AR-1 with positive autocorrelation $\rho \in (0, 1)$. This means that $\Sigma_{ij} = \sigma^2 \rho^{|i-j|}$ and the variance of $\hat{\beta}$ is

$$\frac{\sigma^2}{n^2} \sum_{i,j=1}^n \rho^{|i-j|} \geq \frac{\sigma^2}{n} \sum_{k=0}^{n-1} \rho^k = \frac{\sigma^2(1-\rho^n)}{n(1-\rho)} \gg \frac{\sigma^2}{n}.$$

The last line holds for n large and $\rho \approx 1$.

An even simpler and more extreme example is when the Y_i 's are identical. This corresponds to $\Sigma = \sigma^2 \mathbf{1}\mathbf{1}^\top$ and $\text{Var}(\hat{\beta}) = \sigma^2 \gg \sigma^2/n$.

(c) We could use a year-wise block bootstrap.

Problem 5: Estimating out-of-sample error

Key ideas/tools

- training data cannot be used for validation
- (a) Here the goal is to figure out how well the model selected by cross validation will do on a held-out test set, as measured by R^2 . Hence, we care about out-of-sample performance, whereas the R^2 value obtained by Statistician 1 is in-sample, computing on the same data that the model was fit on.
- (b) Consider splitting the data into ten folds. For each fold, run the entire procedure, including cross-validation, on the in-fold data. Use the resulting model to make predictions on the out-of-fold data. Once we do this for each fold, we get a prediction for each data point, from which we can calculate an R^2 value. This time, we are doing a better job estimating out-of-sample performance.

Problem 6: Estimating a Poisson mixture

Key ideas/tools

- EM algorithm
 - Fisher Information
- (a) Suppose that μ_0 and μ_1 are the two Poisson parameters, and let π be the mixing proportion. Then, the log likelihood is

$$\ell(\mu_0, \mu_1, \pi; Y) = \sum_{i=1}^n \log \left((1 - \pi) e^{-\mu_0} \frac{\mu_0^{Y_i}}{Y_i!} + \pi e^{-\mu_1} \frac{\mu_1^{Y_i}}{Y_i!} \right).$$

- (b) Consider $Z_i \stackrel{\text{i.i.d.}}{\sim} \text{Ber}(\pi)$, and $Y_i | Z_i = z \sim \text{Poi}(\mu_z)$ for $z = 0, 1$. The complete data log likelihood is

$$\begin{aligned} \ell(\mu_0, \mu_1, \pi; Y, Z) &= \sum_{i=1}^n (1 - Z_i) (\log(1 - \pi) - \mu_0 + Y_i \log \mu_0) + Z_i (\log \pi - \mu_1 + Y_i \log \mu_1) \\ &= \log(1 - \pi) \sum_{i=1}^n (1 - Z_i) - \mu_0 \sum_{i=1}^n (1 - Z_i) + \log \mu_0 \sum_{i=1}^n (1 - Z_i) Y_i \\ &\quad + \log \pi \sum_{i=1}^n Z_i - \mu_1 \sum_{i=1}^n Z_i + \log \mu_1 \sum_{i=1}^n Z_i Y_i. \end{aligned}$$

(c) Given parameters estimates $\hat{\theta}^t = (\hat{\mu}_0^{(t)}, \hat{\mu}_1^{(t)}, \hat{\pi}^{(t)})$, we need to compute

$$\hat{\tau}_i^{(t)} = \mathbb{E}_{\hat{\theta}^{(t)}}[Z_i|Y_i] = \mathbb{P}_{\hat{\theta}^{(t)}}[Z_i = 1|Y_i] = \frac{\hat{\pi}^{(t)} e^{-\hat{\mu}_1^{(t)}} (\hat{\mu}_1^{(t)})^{Y_i}}{(1 - \hat{\pi}^{(t)}) e^{-\hat{\mu}_0^{(t)}} (\hat{\mu}_0^{(t)})^{Y_i} + \hat{\pi}^{(t)} e^{-\hat{\mu}_1^{(t)}} (\hat{\mu}_1^{(t)})^{Y_i}},$$

and use this to find the expected complete data log likelihood

$$\begin{aligned} \tilde{\ell}(\mu_0, \mu_1, \pi; Y) &= \mathbb{E}_{\hat{\theta}^{(t)}}[\ell(\mu_0, \mu_1, \pi; Y, Z)|Y] \\ &= \log(1 - \pi) \sum_{i=1}^n (1 - \hat{\tau}_i^{(t)}) - \mu_0 \sum_{i=1}^n (1 - \hat{\tau}_i^{(t)}) + \log \mu_0 \sum_{i=1}^n (1 - \hat{\tau}_i^{(t)}) Y_i \\ &\quad + \log \pi \sum_{i=1}^n \hat{\tau}_i^{(t)} - \mu_1 \sum_{i=1}^n \hat{\tau}_i^{(t)} + \log \mu_1 \sum_{i=1}^n \hat{\tau}_i^{(t)} Y_i. \end{aligned}$$

We can then maximize this with respect to the parameters via

$$\hat{\pi}^{(t+1)} = \frac{\sum_i \hat{\tau}_i^{(t)}}{n}, \quad \hat{\mu}_0^{(t+1)} = \frac{\sum_i (1 - \hat{\tau}_i^{(t)}) Y_i}{\sum_i (1 - \hat{\tau}_i^{(t)})}, \quad \hat{\mu}_1^{(t+1)} = \frac{\sum_i \hat{\tau}_i^{(t)} Y_i}{\sum_i \hat{\tau}_i^{(t)}}.$$

We can use the observed Fisher information matrix $-\ddot{\ell}(\hat{\mu}_0, \hat{\mu}_1, \hat{\pi}; Y)$ to get a variance estimate for $\hat{\pi}$ and hence an asymptotic confidence interval. These confidence intervals are based on the general asymptotic theory for the MLE. An alternative would be use the bootstrap. If we do use the bootstrap, we need to be careful that no label flipping occurs, i.e., we should enforce that $\hat{\mu}_1$ is always larger than $\hat{\mu}_2$.

A note of caution: Let $\tilde{\ell}^T(\mu_0, \mu_1, \pi; Y)$ be the expected complete data log-likelihood in the last iteration T . One may try to use the Hessian of the objective $\tilde{\ell}^T(\mu_0, \mu_1, \pi; Y)$, i.e., $-\nabla^2 \tilde{\ell}^T(\hat{\mu}_0^{(T)}, \hat{\mu}_1^{(T)}, \hat{\pi}^{(T)}; Y)$ as an estimate of the information matrix. This however is **wrong**. It approximates the information matrix of the complete data log-likelihood based on (Z, Y) , which however is larger than the information based on only Y :

$$\mathcal{I}_Y(\theta) = \mathcal{I}_{Z,Y}(\theta) - \mathcal{I}_{Z|Y}(\theta),$$

where e.g., $\mathcal{I}_Y(\theta)$ is the information matrix for estimating θ based on Y (which is what we need for asymptotically valid inference). See Louis [1982] for further elaboration, as well as a method that allows one to (correctly) extract the observed information matrix while running the EM algorithm.

3 Applied 2014: Solution ⁴

Problem 1: Modeling relative intensity of light sources

See Section ??.

Problem 2: Big data linear regression

1. Let $X^1, \dots, X^K \in \mathbb{R}^p$ be the unique values of X_i in the dataset. Then, we can approximate the linear regression model with

$$Y_i = \sum_{\ell=1}^K \mu_{\ell} I(X_i = X^{\ell}) + \varepsilon_i = \mu_{A_i} + \varepsilon_i, \quad \varepsilon_i \stackrel{\text{i.i.d.}}{\sim} N(0, \sigma^2).$$

This is just a cell-means ANOVA model. The MLE for this model is

$$\hat{\mu}_{\ell} = \frac{\sum_{i=1}^n I(A_i = \ell) Y_i}{\sum_{i=1}^n I(A_i = \ell)}, \quad (6)$$

i.e. the averages for each group. To apply this model to new data, we would need the new X values to belong to the set $\{X^1, \dots, X^K\}$. (If the new X values do not belong to this set, we could approximate X by an element in the set $\{X^1, X^2, \dots, X^K\}$ based on some criterion, for example, nearest neighbor in Euclidean distance. One could also consider predicting Y with a weighted average of the $\hat{\mu}_{\ell}$'s where the weights are some monotone decreasing function of the distance between X^l and the new test point.)

2. Let $f_{\mu}(X_i) = \mu_{A_i}$ be the fitted value for X_i . Note that in part 1, we implicitly used the loss function

$$\mathcal{L}(\mu) = \sum_{i=1}^n \frac{1}{n} (f_{\mu}(X_i) - Y_i)^2.$$

which puts equal weight on each observation. If we know that the test set will have different proportions of X_i , then it makes sense to change our loss function to weight data points X_i more heavily if they are more likely to show up in the test set:

$$\mathcal{L}_{\pi}(\mu) = \sum_{i=1}^n \pi_{A_i} (f_{\mu}(X_i) - Y_i)^2.$$

In this example these two loss functions have the same minimizer (6). This is a result of the fact that we are fitting a separate μ for each value of X_i without sharing any information across the different values. Hence, with this model it is

⁴Stefan Wager, Gene Katsevich, Kenneth Tay, Stephen Bates, Nikos Ignatiadis, Isaac Gibbs, D.K.

impossible to take advantage of π to improve the fit on the test data. (Note that if the goal was to predict A from Y rather than to predict Y from A , then the π can be used to improve predictions, but in this setting the goal is to predict Y from A).

3. Suppose that the actual additive model is

$$Y_i = X_i^T \beta + \varepsilon_i, \quad \varepsilon_i \stackrel{\text{i.i.d.}}{\sim} N(0, \sigma^2).$$

Then, β is related to μ via

$$(X^\ell)^T \beta = \mu_\ell, \quad \ell = 1, \dots, K. \quad (7)$$

So, given the true value of μ we can solve for β whenever the $K \times p$ matrix $\tilde{X} = [X^1 \ X^2 \ \dots \ X^K]^T$ has full column rank. If $K > p$, \tilde{X} has full column rank, and we have only an estimate $\hat{\mu}$ of μ it could be the case that the system (7) does not have a solution. In this case we could let $n_l \equiv \sum_{i=1}^n I(A_i = l)$ we could look for $\hat{\beta}$ minimizing $\sum_{\ell=1}^K n_\ell \|(X^\ell)^T \beta - \hat{\mu}_\ell\|_2^2$, i.e. we could take $\hat{\beta} = (\tilde{X}^T \mathcal{W} \tilde{X})^{-1} \tilde{X}^T \mathcal{W} \hat{\mu}$, where $\mathcal{W} = \text{diag}(n_1, n_2, \dots, n_K)$.

On the other hand if $K < p$ is small then (7) is undetermined and there is no unique solution. This situation is comparable to a standard linear model with collinear columns and we could consider fitting β by adding additional regularization. For example, we could let $\hat{\beta}$ be the minimum norm solution to $(X^\ell)^T \hat{\beta} = \hat{\mu}_\ell$, $\ell = 1, \dots, K$. Note that in this case μ is sufficient, namely it completely specifies \hat{Y} , and thus we haven't lost anything by considering μ and we can still solve the OLS problem exactly.

More generally, the fact that the X_i only take on K unique values suggests that we did a bad job parametrizing our regression problem, and that we have many redundant features. We may want to re-think the specification of our regression model, e.g., by deleting or combining redundant features. For example, if the X_i are responses to a survey, we could try summarizing answers to similar questions in a single variable.

Problem 3: Paired versus unpaired regression models

Key ideas/tools:

- correlated errors in linear regression
- mixed effects modeling

Maybe some kids are prone to a sedentary lifestyle, whereas other kids like running around the playground at full speed for the entire duration of recess. Kids in the former

category are likely to have BMI above the mean for both ages, given their treatment status, and kids in the latter category are likely to have BMI below the mean for both ages. This means that ε_{i1} and ε_{i2} are correlated, whereas regression model (1) implicitly assumes they are independent, which means we might get misleading error estimates based on this model. Indeed in the scatter plot we see a very strong correlation between Y_{i1} and Y_{i2} within each of the two treatment status buckets, suggesting that ε_{i1} and ε_{i2} are highly correlated.

A more reasonable model for this data would be model (1) with an extra random effect for each child:

$$Y_{ij} = \beta_0 + \beta_1 X_i + \beta_2 Z_j + \beta_3 X_i Z_j + b_i + \varepsilon_{ij}, \quad b_i \stackrel{\text{i.i.d.}}{\sim} N(0, \sigma_1^2), \quad \varepsilon_{ij} \stackrel{\text{i.i.d.}}{\sim} N(0, \sigma_2^2).$$

The random b_i terms capture fluctuations in children's BMI not explained by age or treatment status.

Note that this model implies that

$$D_i = Y_{i2} - Y_{i1} = \beta_2 + \beta_3 X_i + \varepsilon_i, \quad \varepsilon_i \stackrel{\text{i.i.d.}}{\sim} N(0, 2\sigma_2^2),$$

which is equivalent to model (2). Hence, model (2) is the correct one.

Now, we are used to seeing neglected correlations resulting in *inflated significance* rather than deflated significance, whereas here model (2) is the one with the very low p-value. Why does this happen? The short answer is that if the random effects model holds, but we ignore the random effect, then the random effect will contribute extra variance (σ_1^2) to your coefficient estimates. On the other hand, using the correct model allows us to cancel this random effect and thus we only get the variance from the residual error terms ε_{ij} .

To make the above discussion more transparent, let us consider the model for Y_{ij} for the kids in the control group (i.e. $X_i = 0$):

$$Y_{ij} = \beta_0 + \beta_2 Z_j + b_i + \varepsilon_{ij},$$

and suppose we want to estimate β_2 , which is the BMI gained from t_1 to t_2 . This is the paired samples problem. Note that assuming no random effect (model (1)) makes this an unpaired samples problem. Whether we assume paired or unpaired data, we have

$$\hat{\beta}_2 = \bar{Y}_{.2} - \bar{Y}_{.1}.$$

The difference between the two models lies in the variance of $\hat{\beta}_2$. Under the paired samples model, we have

$$\text{Var}[\hat{\beta}_2] = \frac{2}{n} \sigma_2^2.$$

The model of statistician (2) would consistently estimate the above variance term. On the other hand, under the unpaired model, the residual variance is $\sigma_1^2 + \sigma_2^2$. So the statistician treating the two measurement of the same child as independent, would get a variance estimate of

$$\widehat{\text{Var}}[\hat{\beta}_2] \approx \frac{1}{n} (\sigma_1^2 + \sigma_2^2).$$

The factor ‘2’ that is missing could make inference anti-conservative. However, a large σ_1^2 will offset this and make the ‘wrong method’ also behave even more conservatively.

Finally, just to get a visceral understanding of this situation, suppose that σ_1 is large, β_2 is small, and $\sigma_2 = 0$. This means that every single kid’s BMI increases exactly by the small amount β_2 . However, the variance in baseline BMI is huge. So if you forgot about the pairing and just plotted the histograms of BMI before and after, they would look fairly similar. However, if you accounted for the pairing and took BMI differences, you would get a spike at β_2 .

The moral of the story is that if you have paired data, use a paired t-test instead of an unpaired two-sample t-test. For a similar problem but with discrete data see “Combining 2×2 Contingency Tables” in Miller et al. [1976].

Problem 4: Time series regression

Key ideas/tools:

- data reduction with PCA
 - correlated errors, block bootstrap
1. (a) All features matter as they are each proxies for temperature, so sparsity induced by the LASSO is inappropriate. However, the features are likely highly correlated with each other as they are all proxies for temperature, so low-rank modeling is appropriate.
 - (b) The environmental statistician can use cross-validation. However, we need to be careful to ensure that the different folds are (at least approximately) independent. Because there might be local temporal dependency, it is a good idea to use temporally contiguous folds. Another idea since we are interested in hindcasting would be to reserve the first few years as a validation set and use more recent years as training.

Alternative Solution: Another option is to look for an elbow on a scree plot. However, given that the goal is to create good hindcasts the cross-validation method suggested above is probably a better idea. If you suggested a scree plot in this part you could then suggest using CV in part c.

- (c) Since we have already gone to the effort of finding the optimal number of principal components, we might as well use exactly that number of principal components.
2. The wording of the question is a little vague. To make things a little more precise let $\{\hat{T}_t\}_{1000 \leq t \leq 1849}$ denote our hindcasts. My interpretation of the question is that the goal is to understand the variability in \hat{T}_t with respect to new draws of the

observed data $\{T_t\}_{1850 \leq t \leq 2000}$, $\{P_t\}_{1000 \leq t \leq 2000}$. The obvious thing to do is to run a block bootstrap that re-samples the entire set of training data. Note that when you do this you need to preserve the dependence between T_t and P_t . To do this you should resample the data from times 1850 – 2000 separately from the data from 1000-1849. A downside of this is that it may not correctly model the distribution of your predictions on years just prior to 1850. Namely, this procedure breaks the dependence between the training data $\{(P_t, T_t)\}_{1850 \leq t \leq 2000}$ and the testing points P_t that are close in time to 1850. Unfortunately, I do not see an easy way to avoid this.

Now, if we used a Scree plot to pick the number of principle components in part 1 this bootstrap procedure should work reasonably well. However, issues arise due to the fact that we cross-validated. In particular, we need to re-run our entire fitting procedure, including the cross validation, inside the bootstrap samples. However, due to the presence of repeated samples the folds we get in CV will not be independent and thus the CV error will be biased downwards. As a result we do not expect our bootstrap samples to faithfully replicate what would occur if we sampled an entirely new dataset.

Finding a correction for this issue seems quite difficult. A previous coach suggested to abandon the bootstrap and instead use subsampling. The idea behind subsampling is as follows. For simplicity, suppose we have an i.i.d. sample $X_1, \dots, X_n \sim F$ and we want a confidence interval for $\mu := \mathbb{E}[X_i]$. Since we know that $\sqrt{n}(\bar{X} - \mu) \xrightarrow{D} N(0, \sigma^2)$ we find that it is sufficient to get an estimate of $\sigma^2 := \text{Var}(X_i)$. There are of course many ways to do this. One way is to subsample:

- (a) For $b = 1, \dots, B$: Get a subsample $\{X_i^b\}_{1 \leq i \leq m}$ by sampling *without* replacement from $\{X_i\}_{1 \leq i \leq n}$ and let $\hat{\mu}^b := \frac{1}{m} \sum_{i=1}^m X_i^b$.
- (b) Let $\hat{\sigma}^2$ be the observed variance of $\{\hat{\mu}^b\}_{1 \leq b \leq B}$.
- (c) Return the confidence interval $\hat{C} = [\hat{\mu} - z_{1-\alpha} \sqrt{m/n} \hat{\sigma}, \hat{\mu} + z_{1-\alpha} \sqrt{m/n} \hat{\sigma}]$.

The idea here is that when $n \gg m$, $\{X_i^b\}_{1 \leq i \leq m}$ will act as an i.i.d. sample of size m from F . Critically, we note that since m is smaller than n we have to rescale our variance estimate in step 3 to account for the change in the sample size. Finally, as an aside I remark that you do not need to have asymptotic normality for this procedure to work. In particular, instead of estimating the standard deviation you could estimate quantiles $\hat{q}(\alpha/2)$, $\hat{q}(1 - \alpha/2)$ of $\hat{\mu}^b - \hat{\mu}$ and use the confidence interval $\hat{C} = [\hat{\mu} - \hat{q}(\alpha/2) \sqrt{m/n}, \hat{\mu} + \hat{q}(1 - \alpha/2) \sqrt{m/n}]$

Does subsampling make sense in this question? In our case we have that

$$\hat{T}_t = (P_t^{proj})^T \hat{\beta} = (P_t^{proj})^T \beta^* + (P_t^{proj})^T (\hat{\beta} - \beta^*)$$

where here β^* denotes the true/optimal value of β and P_t^{proj} denotes the projection of P_t onto the lower dimensional space. So, in the limit you may expect that $\hat{T}_t \sim (\mathbb{E}[(P_t^{proj})^T \beta^*], \text{Var}((P_t^{proj})^T(\hat{\beta} - \beta^*)))$. In general, we would expect $\text{Var}((P_t^{proj})^T(\hat{\beta} - \beta^*))$ to scale like the inverse of the sample size. So, if we use subsampling we should probably rescale our estimate of $\text{Var}(\hat{T}_t)$ by m/n , where n is the size of the original sample and m is the size of our subsamples (or alternatively we could re-scale the entire histogram of \hat{T}_t). As before, in order to account for the temporal dependencies we should subsample contiguous blocks of the data.

Finally, note that in session I erroneously discussed subsampling to get a confidence interval for $\hat{\beta}$. This will not work (or at least it won't work without major adjustments) because the number of selected principle components k can change in each of the subsamples and this will drastically change the behaviour of $\hat{\beta}$.

3. (a) If we assume i.i.d. errors and base our p -values and confidence intervals on that assumption, then we will end up getting p -values that are too small and confidence intervals that are too narrow.
- (b) We could either assume a time series model for the errors (e.g. AR(1)), or just use nonparametric methods like the bootstrap described in part 2 to get valid uncertainty estimates.

Problem 5: A latent variable model for soccer

Key ideas/tools:

- EM algorithm
 - hidden Markov models
- (a) Let $X = \{z_{s_m,0}, s_{m,1}, o_m\}_{1 \leq m \leq M}$ be the observed data. Our latent variables are the tiers $Z = \{z_s\}_{1 \leq s \leq S}$ the event $Z_s = j$ corresponds to team s being in tier j . Let $\theta = (\pi, p)$ be the model parameters.

The above model for X and Z naturally factors as $P_\theta(X, Z) = P_\theta(Z)P_\theta(X | Z)$. To write down the complete data likelihood, we need to calculate both of these

factors. We know that $Z_s \stackrel{\text{iid}}{\sim} \text{Multi}(\pi, 1)$. Thus,

$$\begin{aligned}
P_\theta(Z) &= \prod_{s=1}^S \pi_{z_s} \\
&= \prod_{s=1}^S \prod_{j=1}^T \pi_j^{I(z_s=j)} \\
&= \prod_{j=1}^T \prod_{s=1}^S \pi_j^{I(z_s=j)} \\
&= \prod_{j=1}^T \pi_j^{Y_j},
\end{aligned}$$

where $Y_j = \sum_{s=1}^S I(z_s = j)$ is the number of teams assigned to tier j . The term $P_\theta(X | Z)$ factors as a product over matches. Recalling that $o_m = 1$ if and only if team $s_{m,1}$ beats team $z_{s_{m,0}}$ we have,

$$P_\theta(X | Z) = \prod_{m=1}^M P_\theta(s_{m,1} \text{ beats } s_{m,0} | Z)^{o_m} P_\theta(s_{m,0} \text{ beats } s_{m,1})^{(1-o_m)}. \quad (8)$$

Since we have conditioned on the latent tiers Z , we know that if $z_{s_{m,1}} = j$ and $z_{s_{m,0}} = k$, then

$$P_\theta(z_{s_{m,1}} \text{ beats } z_{s_{m,0}} | Z) = p_{jk}, \text{ and } P_\theta(z_{s_{m,0}} \text{ beats } z_{s_{m,1}}) = 1 - p_{jk}.$$

Thus, we can write (8) as a product over $1 \leq m \leq M$ and $1 \leq j, k \leq T$,

$$P_\theta(X | Z) = \prod_{m=1}^M \prod_{k=1}^T \prod_{j=1}^T T p_{jk}^{I(z_{s_{m,1}}=j, z_{s_{m,0}}=k)o_m} (1 - p_{jk})^{I(z_{s_{m,1}}=j, z_{s_{m,0}}=k)(1-o_m)}.$$

Next we split each product over k and j into three parts,

$$\begin{aligned}
&\prod_{k=1}^T \prod_{j=1}^T p_{jk}^{I(z_{s_{m,1}}=j, z_{s_{m,0}}=k)o_m} (1 - p_{jk})^{I(z_{s_{m,1}}=j, z_{s_{m,0}}=k)(1-o_m)} \\
&= \left(\prod_{k=1}^T \prod_{j=1}^{k-1} p_{jk}^{I(z_{s_{m,1}}=j, z_{s_{m,0}}=k)o_m} (1 - p_{jk})^{I(z_{s_{m,1}}=j, z_{s_{m,0}}=k)(1-o_m)} \right) \\
&\times \left(\prod_{k=1}^T p_{kk}^{I(z_{s_{m,1}}=k, z_{s_{m,0}}=k)o_m} (1 - p_{kk})^{I(z_{s_{m,1}}=k, z_{s_{m,0}}=k)(1-o_m)} \right) \\
&\times \left(\prod_{k=1}^T \prod_{j=k+1}^T p_{jk}^{I(z_{s_{m,1}}=j, z_{s_{m,0}}=k)o_m} (1 - p_{jk})^{I(z_{s_{m,1}}=j, z_{s_{m,0}}=k)(1-o_m)} \right).
\end{aligned}$$

Now recall the assumption that $p_{jk} = 1 - p_{kj}$. Thus, $p_{kk} = 1/2$ for all k and the middle term is simply a constant C_m that does not depend on our model parameters θ . Furthermore, we can rewrite the last term as follows,

$$\begin{aligned}
& \prod_{k=1}^T \prod_{j=k+1}^T p_{jk}^{I(z_{s_m,1}=j, z_{s_m,0}=k)o_m} (1 - p_{jk})^{I(z_{s_m,1}=j, z_{s_m,0}=k)(1-o_m)} \\
&= \prod_{k=1}^T \prod_{j=k+1}^T (1 - p_{kj})^{I(z_{s_m,1}=j, z_{s_m,0}=k)o_m} p_{kj}^{I(z_{s_m,1}=j, z_{s_m,0}=k)(1-o_m)} \\
&= \prod_{j=1}^T \prod_{k=1}^{j-1} (1 - p_{kj})^{I(z_{s_m,1}=j, z_{s_m,0}=k)o_m} p_{kj}^{I(z_{s_m,1}=j, z_{s_m,0}=k)(1-o_m)}.
\end{aligned}$$

If we swap j and k , we get

$$\begin{aligned}
& \prod_{k=1}^T \prod_{j=k+1}^T p_{jk}^{I(z_{s_m,1}=j, z_{s_m,0}=k)o_m} (1 - p_{jk})^{I(z_{s_m,1}=j, z_{s_m,0}=k)(1-o_m)} \\
&= \prod_{k=1}^T \prod_{j=1}^{k-1} (1 - p_{jk})^{I(z_{s_m,1}=k, z_{s_m,0}=j)o_m} p_{jk}^{I(z_{s_m,1}=k, z_{s_m,0}=j)(1-o_m)}.
\end{aligned}$$

We thus have

$$\begin{aligned}
& \prod_{k=1}^T \prod_{j=1}^T p_{jk}^{I(z_{s_m,1}=j, z_{s_m,0}=k)o_m} (1 - p_{jk})^{I(z_{s_m,1}=j, z_{s_m,0}=k)(1-o_m)} \\
&= C_m \prod_{k=1}^T \prod_{j=1}^{k-1} p_{jk}^{W_{jkm}} (1 - p_{jk})^{L_{jkm}},
\end{aligned}$$

where

$$\begin{aligned}
W_{jkm} &= I(z_{s_m,1} = j, z_{s_m,0} = k)o_m + I(z_{s_m,1} = k, z_{s_m,0} = j)(1 - o_m) \\
L_{jkm} &= I(z_{s_m,1} = j, z_{s_m,0} = k)(1 - o_m) + I(z_{s_m,1} = k, z_{s_m,0} = j)o_m,
\end{aligned}$$

In words, $W_{jkm} = 1$ if a team in tier j beat a team in tier k in match m and $W_{jkm} = 0$ otherwise. Likewise, $L_{jkm} = 1$ if a team in tier j lost to a team in tier k in match m and $L_{jkm} = 0$ otherwise. Note that $W_{jkm} + L_{jkm} = 1$ if a team in tier k played against a team in tier k and $W_{jkm} + L_{jkm} = 0$ otherwise. Now, taking a product over m , we get

$$\begin{aligned}
P_\theta(X \mid Z) &= \prod_{m=1}^M C_m \prod_{k=1}^T \prod_{j=1}^{k-1} p_{jk}^{W_{jkm}} (1 - p_{jk})^{L_{jkm}} \\
&= C \prod_{k=1}^T \prod_{j=1}^{k-1} p_{jk}^{W_{jk}} (1 - p_{jk})^{L_{jk}},
\end{aligned}$$

where $C = \prod_{m=1}^M C_m$ and

$$W_{jk} = \sum_{m=1}^M W_{jkm}$$

$$L_{jkm} = \sum_{m=1}^M L_{jkm}.$$

That is, W_{jk} is the number of times a team in tier j beat a team in tier k and L_{jk} is the number of times a team in tier j lost to a team in tier j . We can now write down the complete log-likelihood (up to a term that does not depend on the parameters $\theta = (\pi, p)$).

$$\begin{aligned} \ell(\theta; X, Z) &= \log(P_\theta(Z)) + \log(P_\theta(X | Z)) \\ &= \sum_{j=1}^T Y_j \log(\pi_j) + \sum_{k=1}^T \sum_{j=1}^{k-1} W_{jk} \log(p_{jk}) + L_{jk} \log(1 - p_{jk}). \end{aligned}$$

For the E-step we need to calculate the conditional expectation of $\ell(\theta; X, Z)$ at a given iterate $\hat{\theta}$. The conditional distribution of $Z | X$ is tricky to work with since each match introduces dependencies between the teams. However, we are told to not worry about making the E-step computationally efficient. We can thus use Bayes's rule to define and calculate the following,

$$\begin{aligned} \omega_{\vec{z}} &= P_{\hat{\pi}, \hat{p}}(Z = \vec{z} | X) \\ &= \frac{P_{\hat{\pi}, \hat{p}}(Z = \vec{z}) P_{\hat{\pi}, \hat{p}}(X | Z = \vec{z})}{\sum_{\vec{z}'} P_{\hat{\pi}, \hat{p}}(Z = \vec{z}') P_{\hat{\pi}, \hat{p}}(X | Z = \vec{z}')}, \end{aligned} \tag{9}$$

where \vec{z} is an element of $\{1, \dots, T\}^S$. We can then marginalize over the coordinates of \vec{z} to compute

$$P_{\hat{\pi}, \hat{p}}(Z_s = j | X), \quad \text{and} \quad P_{\hat{\pi}, \hat{p}}(Z_{s_1} = j, Z_{s_2} = k | X).$$

Once we have calculated these probabilities, we can use linearity of expectation and calculate the expected complete log-likelihood,

$$\tilde{\ell}(\pi, p) = \sum_{j=1}^T \mathbb{E}_{\hat{\pi}, \hat{p}}[Y_j] \log(\pi_j) + \sum_{k=1}^T \sum_{j=1}^{k-1} \mathbb{E}_{\hat{\pi}, \hat{p}}[W_{jk}] \log(p_{jk}) + \mathbb{E}_{\hat{\pi}, \hat{p}}[L_{jk}] \log(p_{jk}).$$

Maximizing with respect to π and p gives the M-step

$$\hat{\pi}'_j = \frac{\mathbb{E}_{\hat{\pi}, \hat{p}}[Y_j]}{S}$$

$$\hat{p}'_{jk} = \frac{\mathbb{E}_{\hat{\pi}, \hat{p}}[W_{jk}]}{\mathbb{E}_{\hat{\pi}, \hat{p}}[W_{jk}] + \mathbb{E}_{\hat{\pi}, \hat{p}}[L_{jk}]}.$$

We could also add constraints that $p_{jk} \leq p_{j,k+1}$ for all j and k as this would correspond to teams in the top tiers (close to 1), will perform better and better against teams in the bottom tiers (close to T). Since this isn't explicitly in the model, we'll ignore it.

To infer the latent tiers $z_{1:S}$, we can run the EM algorithm until convergence. Then we can use (9) to calculate the joint conditional distribution of $z_{1:S}$ given the observations.

- (b) Cross-validate on matches, making sure that for each fold, the in-fold data do not leave out any teams. For a held-out data point m , we can use the conditional probabilities (9) and the fitted values of \hat{p}_{ij} to estimate the probability $o_m = 1$. We can then use cross-entropy or another loss function to evaluate the fitted model. The number of tiers T will likely be limited by our computational resources. With the description above, the E step has complexity at least S^T . Thus, the time it takes to fit the model grows quickly in S and T .
- (c) Let $z_{s,t}$ be the tier of team s at time t . Then, we might model the evolution of a team's tier as a Markov chain. It is probably the case that there is some continuity of tier across time, so a team probably won't jump from the highest tier to the lowest tier from one time point to the next. A reasonable assumption is that a team can either stay at the same tier, or go up or down by one tier. We can model these jump probabilities as tier-specific or universal across tiers. For example, in the latter model, we would have a transition matrix parametrized by the probability of staying at the same tier. Assuming that the probabilities of jumping up and down (when both are possible) are equal, this determines the entire matrix. Taken together, this is a hidden Markov model where the latent state is the class of each team at each time point, and the emission distribution is the generative model from part (a).

Problem 6: Selective inference brainstorming

Key ideas/tools:

- selective inference
- conditional selective inference
- “Bayes is immune to selection bias”
- data splitting
- bootstrap

The key in the question is that we care about selective inference; i.e., we want to give statistical inference that is valid on the selected genes. Standard confidence

intervals won't work: they have $1 - \alpha$ coverage on all genes, but may have 0 coverage on the selected genes.

A first question is the following: what is our model? One starting point is to treat the X as fixed and consider independent:

$$Y_i \sim \mathcal{N}(\mu_i, \sigma^2), \quad i = 1, \dots, n \quad (10)$$

We write $\mu = (\mu_1, \dots, \mu_n)$. Note that (10) does not make the assumption of linearity. The next question is, what is our target?

Univariate targets: Let us start by first discussing the case of univariate targets. We fit the model (for $j = 1, \dots, p$):

$$Y = \alpha + \beta_j X_j + \epsilon \quad (11)$$

For simplicity let us assume that X_j has been centered and scaled to have mean 0 and $X_j^T X_j = 1$. Then we get

$$\hat{\beta}_j = Y^\top X_j \sim \mathcal{N}(\mu^\top X_j, \sigma^2)$$

Hence we could let our population target be $\beta_j = \mu^\top X_j$. Note that this target makes sense outside the linear model, as long as (10) holds. Even with this simple target, we still face the difficulty of needing to form intervals for the selected targets and can try to apply the 3 approaches discussed. In our discussion we make the simplification that σ is known, so that $t_j = \hat{\beta}_j / \sigma$.

1. **Censoring:** In this case our goal is to form confidence intervals with the following guarantee:

$$\mathbb{P}_{\beta_j} [\beta_j \in CI_j \mid j \in \mathcal{S}] = \mathbb{P}_{\beta_j} [\beta_j \in CI_j \mid t_j \geq c] \geq 1 - \alpha$$

We consider the conditional distribution

$$\mathcal{L}_{\beta_j}(t_j \mid t_j > c) \text{ for each } \beta_j \in \mathbb{R}.$$

Denote this conditional distribution by F_{β_j} . We define the interval

$$\mathcal{I}_{\beta_j} = \left[F_{\beta_j}^{-1} \left(\frac{\alpha}{2} \right), F_{\beta_j}^{-1} \left(1 - \frac{\alpha}{2} \right) \right],$$

with the property that $\mathbb{P}_{\beta_j}[t_j \in \mathcal{I}_{\beta_j} \mid t_j > c] \geq 1 - \alpha$. In other words if we want to test the null hypothesis:

$$H_0 : \beta_j = \tilde{\beta}_j,$$

for a fixed $\tilde{\beta}_j$, then we get a valid conditional test by rejecting when $t_j \notin \mathcal{I}_{\tilde{\beta}_j}$. We get a valid confidence set by inverting the hypothesis test, i.e.:

$$CI_j = \{ \beta_j : t_j \in \mathcal{I}_{\beta_j} \}.$$

Exercise for STATS300: Prove that under the assumptions above, i.e., model (10) with known σ^2 , that the above set is in fact an interval. An issue raised by Yu is that with the above method it could happen that the final CIs contain 0, even though we selected features so as not to include 0. This is true; see Weinstein, Fithian, and Benjamini [2013] for an in-depth discussion of this issue. In a sense, the conditional approach to selective inference requires us to be surprised twice: The first time a feature surprises us, we select it. But e.g., to get a nonzero conditional CI, we need it to surprise us a second time.

2. **Bayes:** Bayes estimation presents a simple cure to selection bias. Given a prior $\pi(\beta_j)$, we can compute posterior credible intervals $\Gamma(t_j)$ for β_j . Then,

$$\mathbb{P}[\beta_j \in \Gamma(t_j) | t_j] \geq 1 - \alpha.$$

Notice that this statement hold *conditionally* on t_j , and so is immune to selection bias. Note 1: Bayes credible intervals are not conditionally valid confidence intervals; but they are still interpretable (and maybe even more interpretable than the former). Note 2: This statement only holds if we have a true prior. If we use a non-informative prior (or a conjugate convenience prior), then selection bias matters. Note 3: If we have lots of data (and get to observe all the $\hat{\beta}_j$, not just the selected ones), then we can estimate prior π by empirical Bayes methods. Options include g -estimation, or using Tweedie’s formula to get the mean/variance of the posterior, cf. Efron [2011].

3. **Data-splitting:** A simple (but perhaps not so powerful) approach to the problem is **data splitting**: select genes on one half of the data, then do inference on the second half. Because we did not use the second half for selection, we don’t have selection bias. Inference on the second half of the data can be done with the bootstrap or with the standard OLS intervals. One issue with the problem at hand is that perhaps our collaborator already chose the set \mathcal{S} of important alleles. If we do data-splitting to select them, then we may get a different set \mathcal{S} , that furthermore depends on the seed of the random number generator that determines the split.
4. **Bootstrap:** Morrison and Simon [2018] propose a bootstrap approach to this problem. Their idea is the following: let us say we seek to form a confidence interval for the coefficient with the largest t_j . If we knew the distribution of $\hat{\beta}_{(1)} - \beta_{(1)}$, then we could directly form intervals for $\beta_{(1)}$. However, since we do not know these, we can proceed as follows: Resample all our data, say by a (X, Y) pairs-bootstrap. In the b -th bootstrap replicate, we compute all $\hat{\beta}_j^b, j = 1, \dots, p$ and then sort them. Let $j(b)$ be the index such that $\hat{\beta}_{j(b)}^b$ is the largest coefficient among the bootstrapped coefficients. Then we pretend that $\hat{\beta}_{j(b)}^b - \hat{\beta}_{j(b)}$ has approximately the same distribution as $\hat{\beta}_{(1)} - \beta_{(1)}$. A caveat of this method is

that it must be seen as a heuristic. Using the bootstrap distribution of $\hat{\beta}_{j(b)}^b - \hat{\beta}_{j(b)}$ is better than completely ignoring selection bias, but also one cannot expect to have any theoretical guarantees for this procedure (e.g., it will be very off in the case where all $\beta_j = 0$).

Multivariate target: What about the multivariate target? Now we posit the model:

$$Y = \alpha + X_S \beta_S + \epsilon \quad (12)$$

Then under (10), a target is:

$$\beta_S = (X_S^\top X_S)^{-1} X_S^\top \mu \quad (13)$$

Let $\beta_{S,j}$ be the coordinate of β_S corresponding to feature j , assuming $j \in S$. Then we seek to form pointwise confidence intervals for the selected $\beta_{S,j}$. A somewhat confusing property of this target is that it depends not only on the feature j , but also on the selected set.

An alternative target is the following: If we are willing to assume that (10) holds but with $\mu_i = \alpha + \gamma^\top X_i$, then we could let our target be γ_S , which is different than (13). Since for the genomic applications $p > n$, one would need to make a lot of further assumptions (e.g., sparsity) to be able to estimate or do inference for γ_S .

Thus we mostly discuss feasible approaches towards estimation of target (13).

1. **Censoring:** In this case, the univariate idea conceptually directly extends to the multivariate target. However, the details become a bit more difficult. We let $\mathcal{L}_{\beta_S}(\hat{\beta}_S | t_j > c \text{ for } j \in S, t_j < c \text{ for } j \notin S)$ be the conditional distribution of $\hat{\beta}_S$. Using this distribution, we can perform hypothesis tests based on $\hat{\beta}_S$. The primary challenge is that this likelihood relies on the entire joint regression vector β , and computing this likelihood may be difficult. However, since we are only interested in pointwise intervals for $\beta_{S,j}$, computation becomes tractable again: Lee and Taylor [2014] carries out the details of the computation; also see Lee et al. [2016] for the same idea in the context of selection by the Lasso.
2. **Bayes:** Under the general model (10) it seems difficult to apply Bayes here; one may need to define a prior on μ so that a posterior will be induced on all possible selective targets β_S . One way to achieve this is if we are willing to assume that the linear model is well-specified (and seek to estimate the regression coefficients in the full linear model or (13)), then we can implement Bayes: for example, we could first assume that the components of β from the full, joint regression are i.i.d. from some distribution which we would fit from the data (Empirical Bayes). We would then compute intervals such that

$$\mathbb{P}[\beta_j \in \Gamma(t_j) | t] \geq 1 - \alpha.$$

3. **Data-splitting:** The approach here directly generalizes to the multivariate case.

4 Applied 2015: Solution ⁵

Problem 1: Exponential family manipulations for the Gamma distribution

(a)

$$\begin{aligned} g(x) &= \exp \{ (v-1) \log(x) - x/\lambda - v \log(\lambda) - \log(\Gamma(v)) \} I(x > 0) \\ &= \exp \left\{ \begin{pmatrix} v-1 \\ 1/\lambda \end{pmatrix}^\top \begin{pmatrix} \log x \\ -x \end{pmatrix} - [v \log(\lambda) + \log(\Gamma(v))] \right\} I(x > 0), \end{aligned}$$

$$\text{so } \eta = \begin{pmatrix} v-1 \\ 1/\lambda \end{pmatrix}, y = \begin{pmatrix} \log x \\ -x \end{pmatrix}, \text{ and } \psi = v \log(\lambda) + \log(\Gamma(v)).$$

(b) Since $\log x$ is the sufficient statistic for $\eta_1 = v-1$, we can compute its expectation by differentiating the log partition function:

$$\mathbb{E}_{v,\lambda}[\log(x)] = \frac{d}{d(v-1)} \psi(v, \lambda) = \frac{d}{dv} (\psi(v, \lambda)) = \log \lambda + \frac{\Gamma'(v)}{\Gamma(v)}.$$

Problem 2: Bias in cross-validation, [Hastie et al., 2009, Ch. 7.10]

Context: As you have seen, this type of question comes up a lot. A good reference to read is Chapter 7.10 of the Elements of Statistical Learning [Hastie, Tibshirani, and Friedman, 2009]. Also Exercise 7.10 therein could be instructive.

- (a) No, it will be biased upwards. This is because the models trained in CV use just 9/10's of the training data, and so will (in general) have a larger error rate than if all the training data was used. The CV estimate would be unbiased for $\hat{C}(x)$'s error if you had 9/10 as much training data **and** you also average over the given training data. In particular, we do not expect the CV estimate to be unbiased for the prediction error of the classifier training on the given training data, but rather to be unbiased for the average prediction error averaged over the observed training data (see section 7.12 in [Hastie, Tibshirani, and Friedman, 2009] and [Bates, Hastie, and Tibshirani, 2022]).
- (b) For any binary classifier \hat{C}_{any} , whether or not it was trained on all the data or any particular subset of the data,

$$\mathbb{P}[\hat{C}_{\text{any}}(X) \neq Y] = \mathbb{E}[\mathbb{E}[I\{\hat{C}_{\text{any}}(X) \neq Y\} | X]] = \mathbb{E}[1/2] = 1/2.$$

⁵Lucas Janson, Gene Katsevich, Kenneth Tay, Stephen Bates, Nikos Ignatiadis, Isaac Gibbs, Dan Kluger and M.H.

Thus the error rate of \hat{C} is $1/2$. The CV estimate will now be unbiased, since that error rate is the same for all classifiers and in particular does not depend on how much training data there is or which subset of the data was used.

- (c) Yes, it will bias the CV estimate downwards. You should add the screening into the procedure so that it is performed within each CV fold. Doing selection on the entire data set introduces dependencies between folds. Thus, the CV estimate will be too optimistic.

Problem 3: Confidence intervals for an exponential

First-order accuracy means that the coverage error of the confidence interval decreases at order $O(1/\sqrt{n})$. Second-order accuracy means that it decreases at order $O(1/n)$.

- (a) Note that y has gamma distribution with shape parameter 1 and scale parameter λ . Hence, $\mathbb{E}[y] = \lambda = SD(y)$. The MLE is \bar{y} (the mean), so $\bar{y} \pm z_{.975}\bar{y}/\sqrt{n}$ is an asymptotically-valid 95% confidence ($z_{.975}$ is the 0.975th quantile of the standard Gaussian).

Remark: The above is a bit of a shortcut though that leverages a CLT for the mean of IID variables, but since it would be a good idea to practice constructing confidence intervals based on the asymptotic theory of the MLE in an exponential family I also present the calculations using that approach below.

To construct confidence intervals for the MLE note that the log-likelihood is

$$l(\lambda) = -n\bar{y}/\lambda - n \log(\lambda).$$

Taking derivatives and setting the log likelihood to zero it is easy to check that $\hat{\lambda} = \bar{y}$ gives the MLE and further

$$\ddot{l}(\lambda) = -\frac{2n\bar{y}}{\lambda^3} + \frac{n}{\lambda^2}.$$

Thus the observed Fisher information is $-\ddot{l}(\hat{\lambda}) = n/\bar{y}^2$, so $(\hat{\lambda} - \lambda) \dot{\sim} N(0, [n/\bar{y}^2]^{-1})$. Recalling, $\hat{\lambda} = \bar{y}$ this gives the confidence interval $\bar{y} \pm z_{.975}\bar{y}/\sqrt{n}$.

- (b) Using shape-scale parametrization: $y \sim \text{Gam}(1, \lambda)$ implies $2y/\lambda \sim \text{Gam}(1, 2) = \chi_2^2$. Summing up over the y_i , we obtain $2n\bar{y}/\lambda \sim \chi_{2n}^2$, so an exact 95% CI is

$$\left[\frac{2n\bar{y}}{\chi_{2n,.975}^2}, \frac{2n\bar{y}}{\chi_{2n,.025}^2} \right],$$

where $\chi_{2n,\alpha}^2$ is the α th quantile of the χ_{2n}^2 distribution.

- (c) **(Out of curriculum.)** We could use the BCa (“bias-corrected and accelerated”) method. Details can be found in Section 11.4 of [Efron and Hastie, 2016].
- (d) **(Out of curriculum.)** I think the ABC bootstrap is what the question is going for. It needs only 1% as much computation as the BCa method. (See [DiCiccio and Efron, 1996] for details.)

Problem 4: Causal inferences based on structural assumptions

Context: Mooij, Peters, Janzing, Zscheischler, and Schölkopf [2016]

First, let us assume that the data is not truncated, i.e. this is a full sample from the joint distribution of (x, y) . Then, it is possible that x causes y , since y appears to be a smooth function of x plus symmetrically distributed noise of constant variance. However, y definitely cannot cause x , since the variance of $x|y$ jumps at the point y_0 when the smooth curve mapping $y \mapsto x$ becomes a one-to-many function.

Now, it is conceivable that we have only received a truncated data set, and that, for example, the domain of x might extend further to the right. However, it is still difficult to imagine a way the data can be extrapolated to allow x to be a smooth function of y plus symmetrically distributed, constant variance noise.

Problem 5: Huber loss connection with L_1 penalization

Context: This problem is quite nice as it combines ideas from convex optimization and robust statistics and appears to be from She and Owen [2011]. Note this is a good example of why you should read the entire question. Doing part (d) first can help with part (b).

- (a) The idea is to allow a few (the ℓ_1 penalty induces sparsity in γ) data points to be corrupted while still basically minimizing squared error. Each γ_i represents how much of an outlier the corresponding y_i is, with $\gamma_i = 0$ interpreted as saying that y_i is not an outlier at all.
- (b) First of all, note that the problem is convex, so we are in good shape. Next, note that for any fixed β , the optimization problem decouples over the vector γ . Specifically, define,

$$r_i = y_i - x_i^\top \beta$$

so that r_i is the i th residual. Then, as a function of γ , our loss function is,

$$\begin{aligned} \ell(\gamma) &= \frac{1}{2} \sum_{i=1}^n (r_i - \gamma_i)^2 + \lambda \sum_{i=1}^N |\gamma_i| \\ &= \sum_{i=1}^N \left(\frac{1}{2} (r_i - \gamma_i)^2 + \lambda |\gamma_i| \right). \end{aligned}$$

The value, γ^* that minimizes the above loss function is given by

$$\begin{aligned}\gamma_i^* &= \underset{\gamma_i}{\operatorname{argmin}} \frac{1}{2}(r_i - \gamma_i)^2 + \lambda|\gamma_i| \\ &= \begin{cases} r_i + \lambda & \text{if } r_i < -\lambda, \\ 0 & \text{if } -\lambda \leq r_i \leq \lambda, \\ r_i - \lambda & \text{if } r_i > \lambda. \end{cases} \\ &= S(r_i; \lambda)\end{aligned}$$

That is, γ_i^* is the *soft-threshold function* of r_i . Furthermore, if γ is fixed, then our model is a simple linear regression problem with response $Y - \gamma$. This suggests the following iterative block descent algorithm,

1. Start with $\hat{\gamma}^0 = 0$.
2. For $t = 0, 1, \dots$,
 - (a) Set $\hat{\beta}^t = (X^\top X)^{-1} X^\top (Y - \hat{\gamma}^t)$.
 - (b) Set $r^t = Y - X\hat{\beta}^t$ and $\hat{\gamma}^{t+1} = S(r^t; \lambda)$.

So one way to solve the optimization problem is with the above iterative algorithm. Another approach is to use part (d) and show that the given optimization problem is equivalent to minimizing the Huber loss. A third approach would be to recast the problem as a (partial) LASSO problem with design matrix $\tilde{X} = [X, I_N] \in \mathbb{R}^{N \times (p+N)}$, parameters $\tilde{\beta} = (\beta, \gamma) \in \mathbb{R}^{p+N}$ and an L^1 penalty on γ . For optimization details see here See <https://www.stat.cmu.edu/~ryantibs/convexopt-F18/lectures/coord-desc.pdf> and <https://scs.hosted.panopto.com/Panopto/Pages/Viewer.aspx?id=03b0a4f5-734f-4f91-974b-a9910130191e>.

- (c) We can apply cross-validation with the criterion being just the residual sum of squares:

$$\sum_{i \in \text{out-of-fold}} \left(y_i - \sum_{j=1}^p x_{ij} \hat{\beta}_j \right)^2.$$

There is one problem with this. Certain out-of-fold data points may be outliers. So it might make sense to use a different loss function to evaluate the errors on such points. One possibility would be to use the Huber loss in cross-validation instead of squared error loss.

Alternatively, if you have prior information about the frequency of outliers, you may just adjust λ to achieve a pre-specified level of sparsity in γ .

- (d) From part (b) we know that for a fixed β , the optimal γ^* is given by

$$\gamma_i^* = S(r_i; \lambda),$$

where $r_i = y_i - x_i^\top \beta$ and $S(r; \lambda)$ is the soft-threshold function defined above. If $|r_i| \leq \lambda$, then $\gamma_i^* = 0$ and

$$\frac{1}{2}(r_i - \gamma_i^*)^2 + \lambda|\gamma_i^*| = \frac{1}{2}r_i^2 = \rho(r_i; \lambda),$$

where $\rho(\cdot; \lambda)$ is the Huber loss function from the question. Furthermore, if $|r_i| > \lambda$, then $S(r_i; \lambda) = r_i - \text{sign}(r_i)\lambda$ and

$$|\gamma_i| = |r_i - \text{sign}(r_i)\lambda| = |r_i| - \lambda.$$

Thus,

$$\begin{aligned} \frac{1}{2}(r_i - \gamma_i^*)^2 + \lambda|\gamma_i^*| &= \frac{1}{2}(\text{sign}(r_i)\lambda)^2 + \lambda(|r_i| - \lambda) \\ &= \lambda|r_i| - \frac{1}{2}\lambda^2 \\ &= \rho(r_i; \lambda). \end{aligned}$$

Thus, with the optimal γ^* , the original objective function can be written as

$$\begin{aligned} \frac{1}{2} \sum_{i=1}^n (y_i - x_i^\top \beta - \gamma_i^*)^2 + \lambda \sum_{i=1}^n |\gamma_i^*| \\ = \sum_{i=1}^n \rho(y_i - x_i^\top \beta; \lambda), \end{aligned}$$

showing that the two problems have the same β solution.

- (e) Criterion (1) can help, but there are likely scenarios where it does not succeed in removing the influence of the far away data point. Let's consider the problem with a fixed λ (i.e. outlier budget). If we have a point whose feature vector is far away from the data, then the straight line fit through the rest of the points might be very far away from the outlying data point (for example this can happen if the true response function is nonlinear but is quite linear locally). Hence, while adding the extra γ term will help, there might not be enough outlier budget to correct for this data point entirely, so the fit through the rest of the data might still be impacted. Moreover, the criterion (1) devotes just as much outlier budget to low leverage outliers as it does to high leverage outliers. Note that larger values of λ correspond to smaller outlier budgets.

A different approach would be to use clustering to identify any outliers (e.g. k -means clustering, and identify points with largest distance to their cluster centroid as outliers). We can then either investigate/remove them or give them lower weight in the optimization. Note that if we only use the x values for k -means then this approach is unsupervised.

Problem 6: Life tables, survival curves and GLMs

Context: Efron and Hastie [2016] Chapter 9

- (a) We first review a commonly used approach to estimate the probability that a client survives past each age: the Kaplan-Meier estimator. Let T denote the (random) survival time. Let $p_j = \mathbb{P}[T < j + 1 | T \geq j]$ be the probability of dying before reaching age $j + 1$ conditional on reaching age j . Then, the survival function (conditional on reaching age 60) is

$$s_i = \mathbb{P}[T \geq i | T \geq 60] = \prod_{j=60}^{i-1} \mathbb{P}[T \geq j + 1 | T \geq j] = \prod_{j=60}^{i-1} (1 - p_j).$$

Now, let d_j be the number of deaths at age j and n_j the number of people still alive at age j . We can estimate $\hat{p}_j = d_j/n_j$

$$\hat{s}_i = \prod_{j=60}^{i-1} (1 - \hat{p}_j) = \prod_{j=60}^{i-1} \left(1 - \frac{d_j}{n_j}\right). \quad (14)$$

This estimator is the Kaplan-Meier estimator

The question is asking us how the actuary computed the survival curve. While the Kaplan-Meier estimator is a natural guess, we should also point out that based on the table this is indeed what the actuary did: the column titled “proportion” has the values of $\hat{p}_i = d_i/n_i$, and the column titled “Survival” has the entries \hat{s}_{i+1} , given by (14).

- (b) We could use logistic regression to model $\text{logit}(p_j)$ quadratically in age:

$$d_j \stackrel{\text{ind}}{\sim} \text{Bin}(n_j, p_j), \quad \log\left(\frac{p_j}{1 - p_j}\right) = \beta_0 + j\beta_1 + j^2\beta_2.$$

Once we get the estimated \hat{p}_j , we can use them in the same survival formula as in part (a):

$$s_i^{\text{quad}} = \prod_{j=60}^{i-1} (1 - \hat{p}_j).$$

To implement this in R, one could use the `glm` function. In particular, suppose each of the first three columns in the table is stored as a vector in R.

```
ageSquared <- age^2
quadFit <- glm(cbind(deaths, number-deaths) ~ age+ageSquared,
               family = binomial)
SurvQuad <- cumprod(1-quadFit$fitted.values)
```

Unfortunately, you will not be able to double check on the exam that this is what the second actuary did to compute the column titled SurvQuad, because you will not have access to R. That being said, this is a reasonable guess based on the problem description and the fact that this actuary has training in GLMs and computed a “quadratically smoothed curve”.

Aside: If you are curious whether the answer to part (b) is the answer that the author of the Qualls question was going for, it was indeed the answer the author had in mind. In particular the procedure in part (b) recovers the entries in the column tiled SurvQuad. See the below screenshot for verification of this:

```
#Copied and pasted data from the pdf file of the Qualls exams
datQ62015_vec <- c(60, 58, 2, .034, .966, .989,61, 61, 0, .000, .966, .978,62, 49, 0, .000, .966, .965,
63, 45, 0, .000, .966, .951,64, 42, 1, .024, .943, .937,65, 28, 0, .000, .943, .921,
66, 46, 2, .043, .902, .904, 67, 32, 0, .000, .902, .886,68, 32, 1, .031, .873, .867,
69, 40, 0, .000, .873, .846,70, 25, 1, .040, .838, .824, 71, 39, 1, .026, .817, .800,
72, 23, 0, .000, .817, .774, 73, 22, 1, .045, .780, .747, 74, 28, 1, .036, .752, .719,
75, 30, 2, .067, .702, .688, 76, 34, 1, .029, .681, .656, 77, 12, 1, .083, .624, .623,
78, 19, 2, .105, .559, .588, 79, 17, 2, .118, .493, .551, 80, 24, 4, .167, .411, .514,
81, 31, 3, .097, .371, .475, 82, 17, 0, .000, .371, .436, 83, 28, 2, .071, .345, .396,
84, 28, 1, .036, .332, .356, 85, 29, 2, .069, .309, .316, 86, 31, 3, .097, .279, .278,
87, 28, 3, .107, .249, .240, 88, 16, 2, .125, .218, .205, 89, 20, 6, .300, .153, .171)

#Some formatting of the data
datQ62015 <- matrix(datQ62015_vec,ncol = 6,byrow = TRUE)
colnames(datQ62015) <- c("age","number","deaths","proportion","Survival","SurvQuad")
age <- datQ62015[, "age"]
number <- datQ62015[, "number"]
deaths <- datQ62015[, "deaths"]

#Checking if answer in part (a) matches columns 4 and 5 of the table
print(mean(round(deaths/number,3)==datQ62015[, "proportion"]))

## [1] 1

print(mean(round(cumprod(1-deaths/number),3)==datQ62015[, "Survival"])))

## [1] 1

#Checking part if the answer in part (b) matches column 6 of the table
age2 <- age^2
fitGLM <- glm(cbind(deaths,number-deaths)-age+age2,family = binomial(link = "logit"))
SurvQuad <- cumprod(1-fitGLM$fitted.values)
print(mean(round(SurvQuad,3)==datQ62015[, "SurvQuad"])))

## [1] 1
```

5 Applied 2016: Solution ⁶

Problem 1: Quadratic discriminant analysis and EM

Key ideas/tools:

- LDA and QDA as Gaussian mixture models
- using EM for semi-supervised learning.

Let $y = 1$ denote “otic” cells and $y = 0$ denote non-otic cells.

- (a) From the left plot, we see that the two Gaussian distributions have different means and covariance matrices. Hence, we can posit the following generative model:

$$\begin{aligned} y_i &\stackrel{\text{iid}}{\sim} \text{Ber}(\pi); \\ x_i | y_i = k &\sim \mathcal{N}(\mu_k, \Sigma_k), \quad k = 0, 1. \end{aligned} \tag{15}$$

To fit these parameters, we can use maximum likelihood, for $k = 0, 1$,

$$\begin{aligned} \hat{\pi} &= \frac{1}{n} \sum_{i=1}^n y_i, \\ \hat{\mu}_k &= \frac{\sum_{i=1}^n x_i I(y_i = k)}{\sum_{i=1}^n I(y_i = k)}, \\ \hat{\Sigma}_k &= \frac{\sum_{i=1}^n (x_i - \hat{\mu}_k)(x_i - \hat{\mu}_k)^\top I(y_i = k)}{\sum_{i=1}^n I(y_i = k)}. \end{aligned} \tag{16}$$

Based on these fitted parameters, we can classify a point x as otic if

$$\log \frac{\hat{\mathbb{P}}[y = 1|x]}{\hat{\mathbb{P}}[y = 0|x]} > 0, \tag{17}$$

where $\hat{\mathbb{P}}$ is the probability distribution under the parameters $\hat{\pi}, \hat{\mu}_k$ and $\hat{\Sigma}_k$. This is quadratic discriminant analysis (QDA).

- (b) We have by Bayes’ rule

$$\begin{aligned} \log \frac{\mathbb{P}[y = 1|x]}{\mathbb{P}[y = 0|x]} &= \log \left(\frac{\pi}{1 - \pi} \frac{\phi_{\mu_1, \Sigma_1}(x)}{\phi_{\mu_0, \Sigma_0}(x)} \right) \\ &= \log \left(\frac{\pi}{1 - \pi} \right) - \frac{1}{2} \log |\Sigma_1| - \frac{1}{2} (x - \mu_1)^\top \Sigma_1^{-1} (x - \mu_1) \\ &\quad + \frac{1}{2} \log |\Sigma_0| + \frac{1}{2} (x - \mu_0)^\top \Sigma_0^{-1} (x - \mu_0). \end{aligned} \tag{18}$$

This quantity is a quadratic form in x , so we get a quadratic decision boundary.

⁶Gene Katsevich, Kenneth Tay, Stephen Bates, Nikos Ignatiadis, Isaac Gibbs, Dan Kluger and M.H.

- (c) I would take the parameter estimates from (16) and plug them into the log-odds expression (18) together with the x value of the new data point. Having calculated this log odds, I would classify the point based on (17).
- (d) Intuitively, the unlabeled data carries information about μ_k and Σ_k . However, we cannot directly use these unlabeled data points without assigning them to classes. To address this problem, we can treat these unknown class labels as missing data, and estimate all the parameters with maximum likelihood estimation by using the EM algorithm. In the E step, we need to calculate the probability

$$\mathbb{P}[z = 1 | x, \pi^{(t)}, \mu_k^{(t)}, \Sigma_k^{(t)}]. \quad (19)$$

This is a calculation similar to (18). In the M step, we would plug these “soft assignments” to calculate the expected complete-data likelihood and maximize with respect to the parameters. One way of initializing the EM algorithm is with the parameter estimates obtained from just the labeled data.

While the problem states “briefly sketch how” and simply mentioning the above could suffice, if there is time on the exam it is worth sketching out in more detail how you would implement the EM algorithm. Let N_l denote the number of labeled samples and N_u denote the number of unlabeled samples y_1, \dots, y_{N_l} are observed indicators of whether each labeled cell is otic and let Z_{N_l+i} for $1 \leq i \leq N_u$ be the unobserved latent variable indicating whether or not the cell is otic. Note that our model for the data is,

$$\begin{aligned} y_i &\stackrel{\text{iid}}{\sim} \text{Ber}(\pi), \\ Z_i &\stackrel{\text{iid}}{\sim} \text{Ber}(\pi), \\ x_i | y_i = k &\sim \mathcal{N}(\mu_k, \Sigma_k), \\ x_i | Z_i = k &\sim \mathcal{N}(\mu_k, \Sigma_k) \end{aligned}$$

Letting $\theta = (\pi, \mu_0, \Sigma_0, \mu_1, \Sigma_1)$ denote the parameter of the model, and observe that the log-likelihood is given by

$$\begin{aligned} \ell(\theta; X, Y) &= \sum_{i=1}^{N_l} y_i \left(\log(\pi) + \log(\phi_{\mu_1, \Sigma_1}(x_i)) \right) + (1 - y_i) \left(\log(1 - \pi) + \log(\phi_{\mu_0, \Sigma_0}(x_i)) \right) \\ &\quad + \sum_{i=N_l+1}^{N_l+N_u} \log(\pi \phi_{\mu_1, \Sigma_1}(x_i) + (1 - \pi) \phi_{\mu_0, \Sigma_0}(x_i)) \end{aligned}$$

We seek to find the $\hat{\theta}$ maximizing the above expression. This can be done by using the EM algorithm, by noting that the complete log likelihood is given by

$$\begin{aligned}
& \ell_c(\theta; X, Y, Z) \\
&= \sum_{i=1}^{N_l} y_i \left(\log(\pi) + \log(\phi_{\mu_1, \Sigma_1}(x_i)) \right) + (1 - y_i) \left(\log(1 - \pi) + \log(\phi_{\mu_0, \Sigma_0}(x_i)) \right) \\
&+ \sum_{i=N_l+1}^{N_l+N_u} Z_i \left(\log(\pi) + \log(\phi_{\mu_1, \Sigma_1}(x_i)) \right) + (1 - Z_i) \left(\log(1 - \pi) + \log(\phi_{\mu_0, \Sigma_0}(x_i)) \right).
\end{aligned}$$

To implement the E-step for each $\hat{\theta}$, i define $\gamma_{\hat{\theta}, i} = y_i$ if $i \in \{1, \dots, N_l\}$ and define

$$\begin{aligned}
\gamma_{\hat{\theta}, i} &:= \mathbb{E}_{\hat{\theta}}[Z_i | X, Y] \\
&= \mathbb{P}_{\hat{\theta}}[Z_i = 1 | x_i] \\
&= \frac{\hat{\pi} \phi_{\hat{\mu}_1, \hat{\Sigma}_1}(x_i)}{\hat{\pi} \phi_{\hat{\mu}_1, \hat{\Sigma}_1}(x_i) + (1 - \hat{\pi}) \phi_{\hat{\mu}_0, \hat{\Sigma}_0}(x_i)}
\end{aligned}$$

For $N_l < i \leq N_l + N_u$,. The expected complete log-likelihood conditional on the observed data is given by

$$\begin{aligned}
& \mathbb{E}_{\hat{\theta}}[l_c(\theta; X, Y, Z) | X, Y] \\
&= \sum_{i=1}^N \gamma_{\hat{\theta}, i} \left(\log(\pi) + \log(\phi_{\mu_1, \Sigma_1}(x_i)) \right) + \left(\log(1 - \pi) + (1 - \gamma_{\hat{\theta}, i}) \log(\phi_{\mu_0, \Sigma_0}(x_i)) \right),
\end{aligned} \tag{20}$$

where $N = N_l + N_u$.

To implement the M-step, note that the above objective is separable in π , (μ_0, Σ_0) and (μ_1, Σ_1) . By taking the derivative with respect to π setting to 0 and by taking the gradients with respect to (μ_k, Σ_k) for $k = 0, 1$ and setting to zero,

$$\pi^{(t+1)} = \frac{1}{N} \sum_{i=1}^N \gamma_{\theta^{(t)}, i}, \quad \mu_1^{(t+1)} = \frac{\sum_{i=1}^N \gamma_{\theta^{(t)}, i} x_i}{\sum_{i=1}^N \gamma_{\theta^{(t)}, i}}, \quad \mu_0^{(t+1)} = \frac{\sum_{i=1}^N (1 - \gamma_{\theta^{(t)}, i}) x_i}{\sum_{i=1}^N (1 - \gamma_{\theta^{(t)}, i})},$$

and

$$\begin{aligned}
\Sigma_1^{(t+1)} &= \frac{\sum_{i=1}^N \gamma_{\theta^{(t)}, i} (x_i - \mu_1^{(t+1)})(x_i - \mu_1^{(t+1)})^\top}{\sum_{i=1}^N \gamma_{\theta^{(t)}, i}}, \\
\Sigma_0^{(t+1)} &= \frac{\sum_{i=1}^N (1 - \gamma_{\theta^{(t)}, i}) (x_i - \mu_0^{(t+1)})(x_i - \mu_0^{(t+1)})^\top}{\sum_{i=1}^N (1 - \gamma_{\theta^{(t)}, i})}.
\end{aligned}$$

Note that solving for $\Sigma_k^{(t+1)}$ in the M-step is a bit tricky and can be done using formulas (57) and (61) in the Matrix cookbook. To find the maximum likelihood you alternate between the E-step and M-step updating $\hat{\theta} = \theta^{(t+1)}$. As mentioned earlier one good way to initialize the EM algorithm here is to use the estimates from part (a).

Problem 2: Three angles and measurement error

Key ideas/tools:

- writing down a model
- maximum likelihood estimation

We find that the angles do not add up exactly to 180° . The problem is that these angles contain measurement error.

In order to estimate the true angles, we must postulate a probability model for the measurement error. If the measurements are y_1, y_2, y_3 and the true angles are $\theta_1, \theta_2, \theta_3$, then the simplest model for measurement error is

$$y_i = \theta_i + \varepsilon_i, \quad \varepsilon_i \stackrel{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2). \quad (21)$$

Assuming a constant error variance would be problematic if the angles were very different in size (e.g. if one of the angles was 1°), but in our case it seems reasonable. We have the additional constraint that $\theta_1 + \theta_2 + \theta_3 = 180$, so we can parameterize the problem using (θ_1, θ_2) only. Hence, we can estimate these two parameters from the following linear regression:

$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 - 180 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ -1 & -1 \end{pmatrix} \begin{pmatrix} \theta_1 \\ \theta_2 \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \end{pmatrix}. \quad (22)$$

Then, $(\hat{\theta}_1, \hat{\theta}_2, 180 - \hat{\theta}_1 - \hat{\theta}_2)$ would be our best guess for the true angles (these should also satisfy the constraints of positive angles, otherwise we could incorporate the constraints into the least squares optimization). The OLS estimate ends up being

$$\begin{bmatrix} \hat{Y}_1 \\ \hat{Y}_2 \\ \hat{Y}_3 \end{bmatrix} = \begin{bmatrix} \frac{2}{3}Y_1 - \frac{1}{3}(Y_2 + Y_3) + 60^\circ \\ \frac{2}{3}Y_2 - \frac{1}{3}(Y_1 + Y_3) + 60^\circ \\ \frac{2}{3}Y_3 - \frac{1}{3}(Y_1 + Y_2) + 60^\circ \end{bmatrix}$$

Problem 3: A missing species problem

Key ideas/tools:

- mixture models
- Empirical Bayes

Context: Efron [2016] proposes empirical Bayes estimation in which the prior is specified as a flexible exponential family. Estimation can proceed parametrically by maximum likelihood. I think the question is not worded very clearly and may be interpreted in multiple ways depending on how you interpret the description of the distribution of X_i . Below we just present one interpretation.

(a) We have

$$\begin{aligned}
 f_k &= \mathbb{P}[X_i = k] = \sum_{j=1}^m \mathbb{P}[X_i = k, \Theta_i = \theta_{(j)}] \\
 &= \sum_{j=1}^m \mathbb{P}[\Theta_i = \theta_{(j)}] \mathbb{P}[X_i = k | \Theta_i = \theta_{(j)}] \\
 &= \sum_{j=1}^m g_j e^{-\theta_{(j)}} \frac{\theta_{(j)}^k}{k!}.
 \end{aligned} \tag{23}$$

Hence,

$$p_{kj} = e^{-\theta_{(j)}} \frac{\theta_{(j)}^k}{k!}. \tag{24}$$

(b) Let $P_k = (p_{k1}, p_{k2}, \dots, p_{km})$. Note that the likelihood for one observation of a species trapped k times

$$\tilde{f}_k(\alpha) = \mathbb{P}[X_i = k | X_i > 0] = \frac{f_k}{1 - f_0} = \frac{P_k^\top \mathbf{g}(\alpha)}{1 - P_0^\top \mathbf{g}(\alpha)}.$$

Therefore the log-likelihood for our count data is given by

$$\begin{aligned}
 \ell(\alpha) &= \sum_{k=1}^{100} y_k \log(\tilde{f}_k(\alpha)) \\
 &= \sum_{k=1}^{100} y_k \log(P_k^\top \mathbf{g}(\alpha)) - \log(1 - P_0^\top \mathbf{g}(\alpha)) \sum_{k=1}^{100} y_k
 \end{aligned}$$

We can then fit $\hat{\alpha}$ by maximum likelihood:

$$\hat{\alpha} = \arg \max_{\alpha} \left(\sum_{k=1}^{100} y_k \log(P_k^\top \mathbf{g}(\alpha)) - \log(1 - P_0^\top \mathbf{g}(\alpha)) \sum_{k=1}^{100} y_k \right)$$

Efron [2016] actually uses a direct second order numerical optimization routine and explicitly calculates gradients (the Score) and Hessians of ℓ w.r.t. α . Another natural way of optimizing this problem is with the EM algorithm. For the EM based approach, let X_1, \dots, X_n be the number of times each observed species is

trapped and let $\Theta_1, \dots, \Theta_n$ be the unobserved latent variable for each species. The complete log-likelihood, is given by

$$\ell_c(\alpha; X, \Theta) = \sum_{i=1}^n \sum_{k=1}^{100} \sum_{j=1}^m I\{\Theta_i = \theta_{(j)}, X_i = k\} \left(\log(\mathbf{g}(\alpha)_j) + \log\left(\frac{p_{kj}}{1 - p_{0j}}\right) \right).$$

To compute the E-step note that

$$\gamma_{\tilde{\alpha}, k, j} \equiv P_{\tilde{\alpha}}(\Theta_i = \theta_{(j)} | X_i = k) = \frac{\mathbf{g}(\alpha)_j p_{kj} / (1 - p_{0j})}{\sum_{j'=1}^m \mathbf{g}(\alpha)_{j'} p_{kj'} / (1 - p_{0j'})},$$

so the expected complete log likelihood (up to an additive constant which doesn't depend on α) is given by

$$\mathbb{E}_{\tilde{\alpha}}[\ell_c(\alpha; X, \Theta) | X] = \sum_{k=1}^{100} \sum_{j=1}^m y_k \gamma_{\tilde{\alpha}, k, j} \log(\mathbf{g}(\alpha)_j).$$

Depending on the function form of \mathbf{g} the M-step could require using Lagrange Multipliers.

- (c) If we had access to the full data (i.e., we also observed $y_0 = \#\{i : X_i = 0\}$), the log-likelihood is given by

$$\ell(\mathbf{h}) = \sum_{k=0}^{100} y_k \log(P_k^\top \mathbf{h}),$$

where $P_k = (p_{k1}, p_{k2}, \dots, p_{km})$ for $k = 0, \dots, 100$ are the fixed vectors defined in part (b). Observe that the log-likelihood is concave in \mathbf{h} , because log is concave, the log composed with a linear function in \mathbf{h} is also concave and hence each $\log(P_k^\top \mathbf{h})$ term is concave in \mathbf{h} . Further all $y_k \geq 0$, so $\ell(\mathbf{h})$ is a positive weighted sum of concave functions in \mathbf{h} and hence $\ell(\mathbf{h})$ is concave.

We can thus estimate \mathbf{h} by minimizing the negative log likelihood $-\ell(\mathbf{h})$ subject to the constraint that $h_i \geq 0$ and $\sum_{i=1}^m h_i = 1$. In particular we can find the \mathbf{h} that solves the following convex optimization problem in \mathbf{h} :

$$\begin{aligned} & \text{minimize} && - \sum_{k=0}^{100} y_k \log(P_k^\top \mathbf{h}) \\ & \text{subject to} && h_i \geq 0, \quad i = 1, \dots, m \\ & && \mathbf{1}^\top \mathbf{h} = 1. \end{aligned}$$

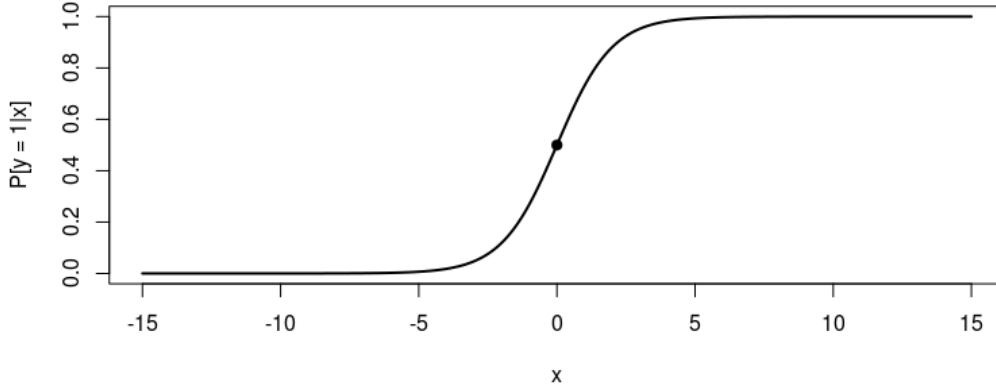


Figure 1: Logistic regression through the origin; $\beta = 1$.

Problem 4: Separation in logistic regression without intercept

Key ideas/tools:

- manipulating the logistic regression likelihood

We have pairs (x_i, y_i) , $i = 1, \dots, n$, that we model as

$$y_i \stackrel{\text{ind}}{\sim} \text{Ber}(p_i), \quad \text{logit}(p_i) = \beta x_i. \quad (25)$$

Because $y_i = 1$ for all i , we get a log-likelihood of

$$\ell(\beta) = \log \prod_{i=1}^n p_i = \sum_{i=1}^n \log \left(\frac{e^{\beta x_i}}{1 + e^{\beta x_i}} \right) = \beta \sum_{i=1}^n x_i - \sum_{i=1}^n \log(1 + e^{\beta x_i}). \quad (26)$$

We know that usually in logistic regression, we have problems when the data are perfectly separated. However, the situation is a bit different here because the regression has no intercept term. With this constraint, the function $\mathbb{P}[y = 1|x]$ must pass through the point $(0, 1/2)$, as in Figure 1. By looking at this figure, it is clear we can fit the data increasingly well by sending $\beta \rightarrow \infty$ if $x_i \geq 0$ for all i . By flipping the sign of β , we can also fit the data increasingly well by sending $\beta \rightarrow -\infty$ if $x_i \leq 0$ for all i . In these cases, we do not get a finite MLE.

On the other hand, if there is at least one $x_i > 0$ and at least one $x_i < 0$, then $|\beta| \rightarrow \infty$ would lead to a log-likelihood tending to negative infinity. Since the log-likelihood is concave in β (we can check that $\ddot{\ell}(\beta) < 0$), this means that it has a unique finite maximizer.

Problem 5: ANOVA – with interactions without

Key ideas/tools:

- calculations with the linear model

Here we present two solutions. The first uses vectorized notation, the second works with the coordinates.

Vectorized solutions

Let $\mathbf{1}_I$ and $\mathbf{1}_J$ represent the vectors of all 1's in dimensions I and J respectively. The “outer product” $\mathbf{1}_I \mathbf{1}_J^\top$ is a matrix in $\mathbb{R}^{I \times J}$ of all 1's. We will think of $Y = (Y_{ij})$ as an element of $\mathbb{R}^{I \times J}$. The given model can be written as

$$Y = \mu \mathbf{1}_I \mathbf{1}_J^\top + \alpha \mathbf{1}_J^\top + \mathbf{1}_I \beta^\top + \gamma + e,$$

where $e \in \mathbb{R}^{I \times J}$ has i.i.d. entries with mean 0 and variance σ^2 . The constraints on α, β and γ can be written in vector notation,

$$\begin{aligned}\alpha^\top \mathbf{1}_I &= 0, \\ \beta^\top \mathbf{1}_J &= 0, \\ \gamma \mathbf{1}_J &= 0, \\ \gamma^\top \mathbf{1}_I &= 0.\end{aligned}$$

In particular, γ is orthogonal to the space of main affects

$$U = \{\mu \mathbf{1}_I \mathbf{1}_J^\top + \alpha \mathbf{1}_J^\top + \mathbf{1}_I \beta^\top\} \subseteq \mathbb{R}^{I \times J}. \quad (27)$$

With this notation we will continue the question.

- (a) In this model, the OLS estimate of μ is the grand mean,

$$\hat{\mu} = \bar{Y}_{+,+} = \frac{1}{IJ} \sum_{i=1}^I \sum_{j=1}^J Y_{i,j} = \frac{1}{IJ} \mathbf{1}_I^\top Y \mathbf{1}_J.$$

The OLS estimate of α is

$$\hat{\alpha}_i = \bar{Y}_{i,+} - \bar{Y}_{+,+},$$

where $\bar{Y}_{i,+}$ is the group mean for level i . That is,

$$\bar{Y}_{i,+} = \frac{1}{J} \sum_{j=1}^J Y_{i,j} = \frac{1}{J} (Y \mathbf{1}_J)_i.$$

We can calculate the expectation of $\bar{Y}_{i,+}$

$$\begin{aligned}\mathbb{E}[\bar{Y}_{i,+}] &= \frac{1}{J} \sum_{j=1}^J \mathbb{E}[Y_{i,j}] \\ &= \frac{1}{J} \sum_{j=1}^J \mu + \alpha_i + \beta_j + \gamma_{ij} + \mathbb{E}[e_{ij}] \\ &= \mu + \alpha_i,\end{aligned}$$

since $\beta_j^\top \mathbf{1}_J = 0$, $\gamma \mathbf{1}_J = 0$ and $\mathbb{E}[e_{ij}] = 0$. By a similar calculation, the expectation of $\bar{Y}_{+,+}$ is μ and thus,

$$\mathbb{E}[\hat{\alpha}_i] = \mathbb{E}[Y_{i,+}] - \mathbb{E}[\bar{Y}_{+,+}] = \mu + \alpha_i - \mu = \alpha_i.$$

Thus,

$$\mathbb{E}[\hat{\alpha}_i - \alpha_{i'}] = \alpha_i - \alpha_{i'},$$

so the OLS estimate for $\alpha_i - \alpha_{i'}$ is unbiased.

- (b) Let $\hat{Y} \in \mathbb{R}^{I \times J}$ be the fitted values in the main effects model. The main effects model has $I + J - 1$ degrees of freedom (there are $I + J + 1$ parameters and 2 constraints). The usual estimate $\hat{\sigma}^2$ is thus

$$\hat{\sigma}^2 = \frac{1}{IJ - (I + J - 1)} \|Y - \hat{Y}\|_F^2 = \frac{1}{(I - 1)(J - 1)} \|Y - \hat{Y}_F\|_F^2$$

where $\|\cdot\|_F$ is the Frobenius norm on $\mathbb{R}^{I \times J}$. Let H be the projection onto the subspace U from (27). We thus have,

$$\begin{aligned}Y - \hat{Y} &= (I - H)Y \\ &= (I - H)(\mu \mathbf{1}_I \mathbf{1}_J^\top + \alpha \mathbf{1}_J^\top + \mathbf{1}_I \beta^\top + \gamma + e) \\ &= \gamma + (I - H)e.\end{aligned}$$

The last line holds because by definition the main effects $\mu \mathbf{1}_I \mathbf{1}_J^\top + \alpha \mathbf{1}_J^\top + \mathbf{1}_I \beta^\top$ are in U and because γ is orthogonal to U . Note that the rank of $(I - H)$ is exactly $IJ - (I + J + 1) = (I - 1)(J - 1)$ and thus,

$$\begin{aligned}\mathbb{E}[\hat{\sigma}^2] &= \frac{1}{(I - 1)(J - 1)} \mathbb{E}[\|\gamma + (I - H)e\|_F^2] \\ &= \frac{1}{(I - 1)(J - 1)} \mathbb{E}[\|\gamma\|_F^2 + 2\gamma^\top (I - H)e + \|(I - H)e\|_F^2] \\ &= \frac{1}{(I - 1)(J - 1)} (\|\gamma\|_F^2 + (I - 1)(J - 1)\sigma^2) \\ &= \sigma^2 + \frac{\|\gamma\|_F^2}{(I - 1)(J - 1)}.\end{aligned}$$

That is $\hat{\sigma}^2$ is biased upwards by,

$$\frac{1}{(I-1)(J-1)} \|\gamma\|_F^2 = \frac{1}{(I-1)(J-1)} \sum_{i=1}^I \sum_{j=1}^J \gamma_{ij}^2 \geq 0.$$

If we make the additional assumption that $e_{ij} \stackrel{\text{iid}}{\sim} \mathcal{N}(0, \sigma^2)$, then we can identify the distribution of $\hat{\sigma}^2$. By the above calculations we have

$$Y - \hat{Y} \sim \mathcal{N}_{I \times J}(\gamma, \sigma^2(I - H)),$$

thus

$$\hat{\sigma}^2 = \frac{1}{(I-1)(J-1)} \|Y - \hat{Y}\|_F^2 \sim \frac{\sigma^2}{(I-1)(J-1)} \chi_{(I-1)(J-1)}^2(\|\gamma\|_F^2).$$

That is $\hat{\sigma}^2$ is a scaled non-central χ^2 distribution.

(c) As seen above

$$\hat{\alpha}_2 - \hat{\alpha}_1 = \bar{Y}_{2,+} - \bar{Y}_{1,+} = \frac{1}{2} (Y_{2,1} + Y_{2,2}) - \frac{1}{2} (Y_{1,1} + Y_{1,2}),$$

is unbiased of $\alpha_2 - \alpha_1$. Thus, we can $\alpha_1 - \alpha_2$ is identifiable.

Coordinate solution

(a) In fact, the usual least squares estimates of α_i are unbiased. Indeed,

$$\begin{aligned} \mathbb{E}[\hat{\alpha}_i] &= \mathbb{E}[\bar{Y}_{i+} - \bar{Y}_{++}] \\ &= \mathbb{E} \left[\frac{1}{J} \sum_{j=1}^J (\mu + \alpha_i + \beta_j + \gamma_{ij} + e_{ij}) - \frac{1}{IJ} \sum_{i=1}^I \sum_{j=1}^J (\mu + \alpha_i + \beta_j + \gamma_{ij} + e_{ij}) \right] \\ &= \mu + \alpha_i - \mu \\ &= \alpha_i \end{aligned} \tag{28}$$

Thus, the usual least squares estimates of $\alpha_i - \alpha_{i'}$ will also be unbiased.

(b) Note that

$$\begin{aligned} \hat{\mu} &= \frac{1}{IJ} \sum_{i,j} Y_{ij} = \frac{1}{IJ} \sum_{i,j} (\mu + e_{ij}) = \mu + \bar{e}_{++}, \\ \hat{\alpha}_i &= \frac{1}{J} \sum_j Y_{ij} - \frac{1}{IJ} \sum_{i,j} Y_{ij} = \alpha_i + \bar{e}_{i+} - \bar{e}_{++}, \\ \hat{\beta}_j &= \frac{1}{I} \sum_i Y_{ij} - \frac{1}{IJ} \sum_{i,j} Y_{ij} = \beta_j + \bar{e}_{+j} - \bar{e}_{++}. \end{aligned}$$

Thus, we have

$$\begin{aligned}
\mathbb{E}[\hat{\sigma}^2] &= \mathbb{E} \left[\frac{1}{(I-1)(J-1)} \sum_{i=1}^I \sum_{j=1}^J (Y_{ij} - \hat{\mu} - \hat{\alpha}_i - \hat{\beta}_j)^2 \right] \\
&= \mathbb{E} \left[\frac{1}{(I-1)(J-1)} \sum_{i=1}^I \sum_{j=1}^J (\gamma_{ij} + e_{ij} - \bar{e}_{i+} - \bar{e}_{+j} + \bar{e}_{++})^2 \right] \\
&= \frac{1}{(I-1)(J-1)} \sum_{i=1}^I \sum_{j=1}^J \mathbb{E} [(\gamma_{ij} + e_{ij} - \bar{e}_{i+} - \bar{e}_{+j} + \bar{e}_{++})^2] \\
&= \mathbb{E} \left[\frac{1}{(I-1)(J-1)} \sum_{i=1}^I \sum_{j=1}^J (e_{ij} - \bar{e}_{i+} - \bar{e}_{+j} + \bar{e}_{++})^2 \right] + \frac{1}{(I-1)(J-1)} \sum_{i=1}^I \sum_{j=1}^J \gamma_{ij}^2 \\
&= \sigma^2 + \frac{1}{(I-1)(J-1)} \sum_{i=1}^I \sum_{j=1}^J \gamma_{ij}^2.
\end{aligned}$$

The last equality holds because the quantity in brackets in the penultimate line is the unbiased estimate for σ^2 in the no-interactions model. Hence, we find that $\hat{\sigma}^2$ is biased upwards for σ^2 . This will have the effect of deflating the significance of F tests for $\alpha_1 = \dots = \alpha_I$ and $\beta_1 = \dots = \beta_J$.

Another way to see that the last line above holds is to use linearity of expectation and to note that since $e_{ij} \stackrel{\text{iid}}{\sim} N(0, \sigma^2)$, then for each i, j ,

$$\begin{bmatrix} e_{ij} \\ \bar{e}_{i+} \\ \bar{e}_{+j} \\ \bar{e}_{++} \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \sigma^2 \begin{bmatrix} 1 & \frac{1}{J} & \frac{1}{I} & \frac{1}{IJ} \\ \frac{1}{J} & \frac{1}{J} & \frac{1}{IJ} & \frac{1}{IJ} \\ \frac{1}{I} & \frac{1}{IJ} & \frac{1}{I} & \frac{1}{IJ} \\ \frac{1}{IJ} & \frac{1}{IJ} & \frac{1}{IJ} & \frac{1}{IJ} \end{bmatrix} \right)$$

and hence letting $v = (1, -1, -1, 1)$,

$$e_{ij} - \bar{e}_{i+} - \bar{e}_{+j} + \bar{e}_{++} \sim \mathcal{N} \left(0, \sigma^2 v^\top \begin{bmatrix} 1 & \frac{1}{J} & \frac{1}{I} & \frac{1}{IJ} \\ \frac{1}{J} & \frac{1}{J} & \frac{1}{IJ} & \frac{1}{IJ} \\ \frac{1}{I} & \frac{1}{IJ} & \frac{1}{I} & \frac{1}{IJ} \\ \frac{1}{IJ} & \frac{1}{IJ} & \frac{1}{IJ} & \frac{1}{IJ} \end{bmatrix} v \right) = \mathcal{N} \left(0, \sigma^2 \left(1 - \frac{1}{J} - \frac{1}{I} + \frac{1}{IJ} \right) \right),$$

so indeed

$$\frac{1}{(I-1)(J-1)} \sum_{i=1}^I \sum_{j=1}^J \mathbb{E} [(e_{ij} - \bar{e}_{i+} - \bar{e}_{+j} + \bar{e}_{++})^2] = \sigma^2.$$

(c) A direct way of checking identifiability is to note that

$$\mathbb{E} \left[\frac{Y_{11} + Y_{12}}{2} - \frac{Y_{21} + Y_{22}}{2} \right] = \alpha_1 - \alpha_2.$$

Problem 6: Testing linearity with replicates

Key ideas/tools:

- F-test
- residual bootstrap

(a) Given i.i.d. normal errors, we can use the F test to compare

$$H_0 : y_{ij} = \beta_0 + \beta_1 x_i + e_{ij} \quad \text{vs.} \quad H_1 : y_{ij} = \sum_{i'=1}^k \beta_{i'} I(x_i = x_{i'}) + e_{ij}. \quad (29)$$

Here, H_0 represents the linear model and H_1 represents the saturated model. The resulting F statistic is

$$F = \frac{\frac{1}{k-2} \sum_{i,j} (\bar{y}_i - \hat{\beta}_0 - \hat{\beta}_1 x_i)^2}{\frac{1}{km-k} \sum_{i,j} (y_{ij} - \bar{y}_i)^2}. \quad (30)$$

This is the ratio of the “lack of fit sum of squares” to the “pure error sum of squares,” and has a null distribution of $F_{k-2, km-k}$.

(b) The main idea here is to describe some sort of bootstrap test. However, there are many possible ways to set this up. Here are the solutions that have accumulated over the years.

Solution 1: Suppose that $y_{ij} = m(x_i) + e_{ij}$, where $e_{ij} \stackrel{\text{i.i.d.}}{\sim} F$ for some unknown error distribution F with mean 0. A reasonable approach here is to bootstrap residuals. In particular let $\hat{e}_{ij} = y_{ij} - \hat{\beta}_0 - \hat{\beta}_1 x_i$. Then, we can generate bootstrap datasets by setting

$$y_{ij}^* = \hat{\beta}_0 + \hat{\beta}_1 x_i + \hat{e}_{ij}^*$$

where \hat{e}_{ij}^* are sampled with replacement from the set $\{\hat{e}_{ij}\}$. We can then recalculate the F statistic (30) for each bootstrapped data set and thereby generate its null distribution.

I think it is instructive to also consider the solution of a previous coach who suggested to use the same procedure, but with $\hat{e}_{ij} = y_{ij} - \bar{y}_i$. Note that in this case $\hat{e}_{ij} = e_{ij} - \bar{e}_i$. If m is small the empirical distribution of \hat{e}_{ij} may not be very close to F and thus we probably should not expect this procedure to work. On the other hand, if m is large this is likely also valid.

If we think that the error distribution changes with x (e.g. if there is heteroskedasticity), then we can also resample the residuals separately for each x . However, this will also probably only work when m is large.

A previous coach also suggested that when m is large, we may still trust using the F-test from part (a), if m is large enough for the Central Limit Theorem has kicked in. Typically we expect the Central Limit Theorem to kick in whenever $n \gg p$ or equivalently in this case when $m \gg 1$. However, while I see how the Central Limit Theorem can kick in for the numerator of the F -statistic as $m \rightarrow \infty$, I don't quite see how in the limit as $m \rightarrow \infty$ the central limit theorem would kick in for the denominator and make it approximately χ_{km-k}^2 .

Alternative solutions when m is fixed $k \rightarrow \infty$: An alternative solution is to explicitly evaluate the limiting distribution of the F-statistics. Let $X \in \mathbb{R}^{mk \times 2}$ be the design matrix with rows X_1, \dots, X_{mk} . Write

$$F = \frac{km - k}{k - 2} \left(\frac{y^\top (I - X(X^\top X)^{-1} X^\top) y}{\sum_{i,j} (e_{ij} - \bar{e}_{i\cdot})^2} - 1 \right) = \frac{km - k}{k - 2} \left(\frac{e^\top (I - X(X^\top X)^{-1} X^\top) e}{\sum_{i,j} (e_{ij} - \bar{e}_{i\cdot})^2} - 1 \right)$$

Without loss of generality assume that the second column of X is centred and both columns are scaled so that $X^\top X = I$. Then,

$$\begin{aligned} F &= \frac{km - k}{k - 2} \left(\frac{e^\top e - \|\sum_{i,j} x_i e_{ij}\|^2}{\sum_{i,j} (e_{ij} - \bar{e}_{i\cdot})^2} - 1 \right) \\ &= \frac{km - k}{k - 2} \left(\frac{m \sum_i (\bar{e}_{i\cdot})^2}{\sum_{i,j} e_{ij}^2 - (\bar{e}_{i\cdot})^2} \right) - O_P(1/k) \\ &= \frac{km - k}{k - 2} \left(\frac{m^{-1} \sum_{i,j} e_{ij}^2 + m^{-1} \sum_i \sum_{j \neq j'} e_{ij} e_{ij'}}{(1 - 1/m) \sum_{i,j} e_{ij}^2 - (1/m) \sum_{j \neq j'} e_{ij} e_{ij'}} \right) - O_P(1/k) \end{aligned}$$

So, we find that

$$\begin{aligned} \sqrt{k} \left(F - \frac{k}{k - 2} \right) &= \sqrt{k} \frac{k}{k - 2} \left(\frac{m^{-1} \sum_{i,j} e_{ij}^2 + m^{-1} \sum_i \sum_{j \neq j'} e_{ij} e_{ij'}}{m^{-1} \sum_{i,j} e_{ij}^2 - m^{-1} (m - 1)^{-1} \sum_{j \neq j'} e_{ij} e_{ij'}} - 1 \right) - O_P(1/k) \\ &= \sqrt{k} \frac{k}{k - 2} \left(\frac{(m - 1)^{-1} \sum_i \sum_{j \neq j'} e_{ij} e_{ij'}}{m^{-1} \sum_{i,j} e_{ij}^2 - m^{-1} (m - 1)^{-1} \sum_{j \neq j'} e_{ij} e_{ij'}} \right) - O_P(1/k) \\ &\xrightarrow{D} N(0, 2m(m - 1)) \end{aligned}$$

Note that the limiting distribution does not depend on the distribution of the errors and in particular the limiting distribution when $e_{ij} \sim N(0, \sigma^2)$ is the same as the limiting distribution in the case where $e_{ij} \sim F$ for F an arbitrary distribution with a finite second moment. Thus, the F-test proposed in part a will be accurate without needing to assume normality.

- (c) If there had been no replication, then the denominator of the F statistic would be zero, i.e. the F test would not be valid. Hence, we need to make our saturated

model a little less saturated. For example, if Engineer A sees a polynomial-looking relationship between y and x , then one could do an F test versus a larger model with some higher-order terms in x . Bootstrapping residuals could also work in this case.

If we have densely spaced covariate values x_i , another approach we could take is to approximate the lack of fit F test by simply grouping points with similar x_i . In general, this approach will underestimate the difference between $x_i^\top \beta$ and $m(x_i)$ and thus will produce a conservative test.

- (d) We would question the independence of errors across i if i indexed space or time. We know that spatial and temporal data often exhibit local correlations in errors. We would question the independence of e_{ij} across j if there were systematic differences in the data collection process across i , e.g. if i indexes days and a different person collected the data on each day. This would introduce positive correlations among e_{ij} for each i , which would inflate our F statistics by underestimating their denominators.

6 Applied 2017: Solution ⁷

Problem 1: Bayesian Logistic Regression

Key ideas/tools:

- setting up hierarchical models
- the connection between the posterior mode and penalized estimation

(a) The model is (treating x_{ij} as fixed)

$$\beta \mid \tau \sim \mathcal{N}(0, \tau^2 I_p), \quad y_{ij} \mid \beta \stackrel{ind}{\sim} \text{Ber}(p_{ij}), \text{ where } \text{logit}(p_{ij}) = x_{ij}^\top \beta. \quad (31)$$

(b) In this set-up, we are assuming that X is fixed. Then the posterior is

$$\begin{aligned} p(\beta \mid y) &\propto p(\beta)p(y \mid \beta) \\ &\propto \exp\left(-\frac{1}{2\tau^2}\|\beta\|^2\right) \cdot \prod_{i,j} \left[\left(\frac{\exp(x_{ij}^\top \beta)}{1 + \exp(x_{ij}^\top \beta)} \right)^{y_{ij}} \left(\frac{1}{1 + \exp(x_{ij}^\top \beta)} \right)^{1-y_{ij}} \right] \\ &= \exp\left(-\frac{1}{2\tau^2}\|\beta\|^2\right) \cdot \prod_{i,j} \frac{[\exp(x_{ij}^\top \beta)]^{y_{ij}}}{1 + \exp(x_{ij}^\top \beta)}. \end{aligned}$$

Thus, the posterior distribution can be written as

$$p(\beta \mid y) = \frac{\exp\left(-\frac{1}{2\tau^2}\|\beta\|^2\right) \cdot \prod_{i,j} \frac{[\exp(x_{ij}^\top \beta)]^{y_{ij}}}{1 + \exp(x_{ij}^\top \beta)}}{\int_{\mathbb{R}^p} \exp\left(-\frac{1}{2\tau^2}\|\beta\|^2\right) \cdot \prod_{i,j} \frac{[\exp(x_{ij}^\top \beta)]^{y_{ij}}}{1 + \exp(x_{ij}^\top \beta)} d\beta}. \quad (32)$$

Computing the posterior mode of β —also called the MAP (Maximum a-posteriori) estimator—amounts to minimizing

$$-\log p(\beta \mid y) = \frac{1}{2\tau^2}\|\beta\|^2 - \sum_{i,j} [y_{ij}x_{ij}^\top \beta - \log(1 + \exp(x_{ij}^\top \beta))]. \quad (33)$$

Notice the posterior mode is exactly equivalent to the maximizer of an L_2 -penalized (Ridge) logistic regression. This is a convex problem and e.g., Newton-Raphson will converge rapidly.

To estimate the posterior expectation of β (or any function of the posterior distribution of β), we can use an MCMC algorithm, such as the Metropolis-Hastings (MH) algorithm. (The main idea with these algorithms is that we try to construct a Markov chain whose stationary distribution is the posterior distribution

⁷Kenneth Tay, Stephen Bates, Nikos Ignatiadis, Isaac Gibbs, D.K.

we want to sample from.) The algorithm gives us samples $\beta_1^s, \dots, \beta_n^s$ from the posterior distribution of β . We can then estimate $\mathbb{E}[\beta \mid y]$ with $\frac{1}{n} \sum_{i=1}^n \beta_i^s$.

A simple way of doing MCMC here is with the independence MH algorithm using the fact that a Gaussian approximation is often quite good for the posterior in logistic regression. Concretely:

$$\beta \mid y \sim \mathcal{N}(\tilde{\beta}, \tilde{H}),$$

where a good choice of $\tilde{\beta}$ is the optimizer of (33). For \tilde{H} we can use the inverse of the negative Hessian of (33) evaluated at the minimizer.

The independence MC algorithm then generates samples as follows (we write $q(\beta)$ is the density of $\mathcal{N}(\tilde{\beta}, \tilde{H})$):

- (a) Start with β_0 .
- (b) Draw $\beta \sim q$.
- (c) Let $\alpha = \min \{1, (\pi_{\text{post}}(\beta)/q(\beta))/(\pi_{\text{post}}(\beta_0)/q(\beta_0))\}$, where $\pi_{\text{post}}(\beta) = p(\beta|y)$ is given by (32)).
- (d) Let $\beta_1 = \beta$ with probability α , else $\beta_1 = \beta_0$.
- (e) Repeat for $j = 2, \dots$

Rossi, Allenby, and McCulloch [2005, Chapter 3] recommend using a multivariate t-distribution with say 6 degrees of freedom and mean/covariance as above instead of the Normal proposal.

Note that other answers besides the Independence MH can give full credit. Standard Metropolis-Hastings algorithms where the proposal distribution $q(\cdot)$ is centered about the current value β_j , would also be a reasonable way to estimate the posterior mean (as long as the proposal distributions aren't too narrow or wide, but this can be checked with trace plots). What about **Gibbs** sampling? For Probit regression this can be done (i.e., with a probit instead of logit link) by data augmentation, cf. Albert and Chib [1993]. Coming up with such an algorithm for the logit link turned out to be elusive until Polson, Scott, and Windle [2013].

- (c) One possible hierarchical model is

$$\begin{aligned} \sigma &\sim \text{HalfCauchy}(V) \\ \beta &\sim \mathcal{N}(0, \tau^2 I_p), \\ \beta_i | \beta &\stackrel{iid}{\sim} \mathcal{N}(\beta, \sigma^2 I_p), \\ y_{ij} | \beta_i &\stackrel{ind}{\sim} \text{Ber}(p_{ij}), \quad \text{logit}(p_{ij}) = x_{ij}^\top \beta_i. \end{aligned}$$

Our random coefficients are β_i , and the prior hyperparameters are β and σ . β and σ themselves have hyperparameters τ and V respectively, which are assumed to be known.

A hierarchical model makes sense as we can think of the players in our sample being drawn from some population players; by setting up a hierarchical model, we are able to pool strength across data from different players to obtain better estimates for our model.

It may be useful to note that an alternative non-Bayesian approach to this modelling problem would be to use a random effects model. This would essentially take the same form as the hierarchical model above except that now β and σ would be treated as fixed parameters that need to be estimated.

- (d) For each player i , by looking at the posterior distribution of β_i , we can do inference on how the various covariates affect their probability of making a shot. If player i is on the opponent's team, they can see which covariates x_{ij} are most important to i 's probability of scoring; they can then plan their defensive strategies to influence those particular covariate values. If player i is on the opponent's team, they can try to improve the values of those x_{ij} which matter.

We expect players who are 1) most different from the average player and 2) for whom we do not have a lot of data (i.e., they took only a few shots) to have the largest differences in β_i . This is because they will experience greater shrinkage.

Problem 2: Combining p-values

Key ideas/tools:

- Multiple testing
- False positives are binomially distributed when tests are independent

- (a) Let X_1, \dots, X_{100} be the readings at the 100 sample points.

We are not told whether we should do a one-sided or two-sided test. By default one would always do a two-sided test, but one-sided is also plausible in this problem (e.g., if the investigators are only interested in detecting events that lead to the emission of many photons). If you were consulting, you would ask your collaborators to clarify. For the qual, you can pick either (or both); as long as you provide a quick justification!

Let us do one-sided tests: Then, the smallest p-value would correspond to the receptor that measured $X_i = 20$ photons. The p-value is:

$$P_i = \mathbb{P}[\text{Poisson}(10) \geq 20] = 1 - 0.9965 = 0.0035$$

Since we are testing 100 hypotheses, however, we need to apply a multiple testing correction. With a Bonferroni correction at $\alpha = 0.05$ we would for example only

reject hypotheses with $P_i \leq 5 \cdot 10^{-4}$, and so indeed we would not reject anything and report back to the investigators that no receptor ended up being significant. Unfortunately, other corrections, such as Benjamini–Hochberg, also would lead to no significant results at the 0.05 level, see R code below:

```
Xs <- seq(0,20)
Ns <- c(0,0,1,0,3,2,9,7,11,11,8,8,13,7,5,5,2,3,1,3,1)
pvals <- rep(1 - ppois(1:21 - 1, 10, Ns[1:21]))
adjusted_pvals <- p.adjust(pvals, method="BH")
min(adjusted_pvals) #0.08635855
```

- (b) For each i , let $R_i = 1$ if we reject the null $X_i \sim \text{Pois}(10)$ at the 5% level, $R_i = 0$ otherwise. Let $T = \sum_{i=1}^{100} R_i$. We assume that R_i are independent; this is plausible

since the receptors are placed at “widely scattered” points and so, under the global null, $T \sim \text{Binom}(100, 0.05)$ (In the setting of this problem we know the exact null distribution but in other applied qual problems we may only know T is stochastically smaller than $\text{Binom}(100, 0.05)$). Based on the table on the right, we reject the global null at the 5% level if $T > 9$.

Let us check how many rejections we got at $\alpha = 0.05$: $X_i = 16$ would lead to (1-sided) p-value of $1 - 0.9513 = 0.0487$, while $X_i = 15$ would lead to a p-value > 0.05 . Thus, $R_i = \#\{i : X_i \geq 16\} = 2 + 3 + 1 + 3 + 1 = 10$ and we can (barely) reject the global null hypothesis⁸.

What does this tell us? It provides evidence that at least one of the 100 receptors detected “something”. In this sense, it counts and could be helpful information for the investigators, e.g., if they want to continue investigating.

On the other hand, from a multiple testing perspective, it does not count: we still cannot tell which receptor is the culprit (“detection” is easier than “localization”); at least if we are interested in the guarantee that with high probability we make no false discovery. If, after looking at the data, we pick the most significant values and ask whether they are truly significant or not, we need to account for selection effects through some p -value adjustment.

Note that in a situation as the present one, an intermediate approach would be to try to ‘pool’ receptors, say if we can group them according to geographic regions. Then we could compute a (combined) p-value for each group of receptors and perhaps one of the groups would come up as significant. This would be a starting point for the investigators to pinpoint what happened.

⁸Note that if we had computed 2-sided p-values, then we would not have rejected the global null hypothesis.

Problem 3: Disentangling Days of the Week

Key tools/ides:

- formulating the problem as a linear model

First, let's focus on estimating the fraction of weekly sales attributable to each day of the week for a particular country and a particular retail segment (i.e. online or retail). Let the weekly sales be denoted by W_1, \dots, W_{156} , and let the monthly sales be denoted by M_1, \dots, M_{36} .

Let p_1, \dots, p_7 denote the fraction of weekly sales attributable to Monday, \dots , Sunday. Note that for each monthly sales figure, we can determine which weeks fell completely within that month. This allows us to subtract the sales for those weeks, and we can model the residual sales figure as a linear combination of the p_i 's. To illustrate, assume that M_1 has 30 days, and that it starts on a Wednesday and ends on a Thursday. Then we can model

$$M_1 - W_2 - W_3 - W_4 = (p_3 + \dots + p_7)W_1 + (p_1 + \dots + p_4)W_5 + \varepsilon. \quad (34)$$

We can write such equations for each of the 36 months, resulting in a linear regression model that we can fit by least squares. Note that we could also incorporate the constraints that the p_i lie on the probability simplex, i.e., $\sum p_i = 1$ and $p_i \geq 0$.

One possible limitation of this model is that months where the first or last has a major holiday could be outliers (e.g. stores may be closed on New Year's day). If your manager wants to know the fraction of weekly sales attributable to each day of the week in a typical week, then you may want to remove months for which the first or last week contains a holiday, but it sounds like you are being asked to estimate on average the fraction of weekly sales attributable to each week in which case, holidays should count towards that average.

What about inference and comparison across countries/retail segments? Let us say we want to compare $p_{1,i}$ vs $p_{1,j}$, where i, j correspond to two different country/retail segment combinations. The procedure described above provides us with a point estimate:

$$\widehat{\Delta p_1} = \widehat{p}_{1,i} - \widehat{p}_{1,j}.$$

To conduct inference, we could try to estimate the standard error of $\widehat{\Delta p_1}$. One way to do this would be to use a block bootstrap on the regression by month. This should help account for the dependencies both between months that have overlapping weekly data and due to autocorrelations that exist in the daily data.

More precisely, let $\{(X_t^i, Y_t^i)\}_{1 \leq t \leq T}$ and $\{(X_t^j, Y_t^j)\}_{1 \leq t \leq T}$ denote the covariate response pairs for the i_{th} and j_{th} country/retail segments formed using (34). Then, in a block bootstrap we would split the time frame $1 \leq t \leq T$ into B contiguous blocks. In each bootstrap sample we could re-sample B of the blocks of time with replacement to get new bootstrap samples for the sales in the i_{th} and j_{th} segments. Note that here

we use the **same** times for both the i_{th} and j_{th} segments. This will help preserve any dependencies between the sales in the two segments. In each bootstrap sample we could then re-compute the statistic $\widehat{\Delta p_1}^*$ and then form a bootstrap confidence interval for Δp_1 by looking at the quantiles of $\widehat{\Delta p_1}^* - \widehat{\Delta p_1}$.

Problem 4: Ranking Sports Teams

Key tools/ides:

- formulating the problem as a GLM
- (a) This is a possible approach, in that we expect good teams to obtain higher scores than bad teams. However, it does have a number of weaknesses:
- Obtaining a high score is not the objective of a game; winning is. While the two are correlated, they are not the same.
 - It does not take the strength of the opponent into account. Obtaining a high score against a strong opponent should count for more than obtaining a high score against a weak opponent. While the approach would be reasonable if each team played all of the other teams the same number of times per season, but we are told that each team only plays a subset of the other teams each season.
- (b) Let there be a total of n games and p teams. Let β_j be the strength of team j , where a larger value of β_j represents a stronger team. For each game i , let $Y_{i,\text{home}(i)}$ and $Y_{i,\text{away}(i)}$ be the number of points the home and away teams scored respectively. One possible model is to only model which team won and use the Bradley-Terry model:

$$\text{logit} \left(\mathbb{P} [Y_{i,\text{home}(i)} > Y_{i,\text{away}(i)}] \right) = \beta_{\text{home}(i)} - \beta_{\text{away}(i)}. \quad (35)$$

Note that the model as stated is actually unidentifiable: shifting all the β_i 's by a constant c gives the same fit. To make it identifiable, we could set $\beta_j = 0$ for some j .

An alternative is to more directly model the scores. For example, we could use an cumulative logit model for the values of $Y_{i,\text{home}(i)} - Y_{i,\text{away}(i)}$. e.g. suppose $Y_{i,\text{home}(i)} - Y_{i,\text{away}(i)} \in \{-k, -k+1, \dots, k-1, k\}$. Then, we could use the model

$$\forall -k \leq j \leq k-1, \text{logit} \left(\mathbb{P}(Y_{i,\text{home}(i)} - Y_{i,\text{away}(i)} \leq j) \right) = \alpha_j + \beta_{\text{away}(i)} - \beta_{\text{home}(i)}.$$

This could work well in low-scoring games that have only a few possible values for $Y_{i,\text{home}(i)} - Y_{i,\text{away}(i)}$. One should note that the current parametrization of this

model does not treat the home and away teams symmetrically. If we want to treat the two teams symmetrically we should require that

$$\text{logit}(\mathbb{P}(Y_{i,\text{away}(i)} - Y_{i,\text{home}(i)} \leq j)) = \alpha_j + \beta_{\text{home}(i)} - \beta_{\text{away}(i)}.$$

In our current model we have that

$$\begin{aligned} \text{logit}(\mathbb{P}(Y_{i,\text{away}(i)} - Y_{i,\text{home}(i)} \leq j)) &= \text{logit}(\mathbb{P}(Y_{i,\text{home}(i)} - Y_{i,\text{away}(i)} > -j - 1)) \\ &= -\alpha_{-j-1} + \beta_{\text{home}(i)} - \beta_{\text{away}(i)}. \end{aligned}$$

So to enforce the desired symmetry we should require that $-\alpha_{-j-1} = \alpha_j$ for all j .

- (c) We can introduce a parameter α for “home-field advantage”. For example for the Bradley-Terry model:

$$\text{logit}(\mathbb{P}[Y_{i,\text{home}(i)} > Y_{i,\text{away}(i)}]) = \alpha + \beta_{\text{home}(i)} - \beta_{\text{away}(i)}.$$

Alternatively, in the cumulative logit model we could remove the symmetry constraint.

- (d) Consider the undirected graph where each team is a node and we draw an edge between teams if they have played each other. Fitting, e.g., (35) is a problem if this graph has more than 1 connected component, since again we have the same identifiability problem discussed earlier: We can shift the “ability” β of the teams in each component by an arbitrary amount, and the model would be the same. To get identifiability, we would need to anchor a team in each component, to e.g., $\beta_j = 0$. Since this anchoring is arbitrary, we can however not compare coefficients across components: If Duke is in the same connected component as Stanford, then we could compare their coefficients, otherwise not.

If we *really* want to try to make a guess about Stanford Vs Duke and they are not in the same connected component, this could perhaps be possible if we are willing to assume that two teams j, j' across the two components have the same skill. Then we could set these two teams as the baseline in each component.

Problem 5: Do I Need to Use Bonferroni?

Key ideas/tools:

- different tests have different purposes
- p-values must be interpreted in context

Assume we are testing hypotheses H_1, \dots, H_n at the same level α , say $\alpha = 0.05$. Then applying the Bonferroni correction controls the family-wise error rate (FWER),

i.e. the probability of at least 1 false rejection is $\leq \alpha$. FWER is fundamentally a conservative criterion, protecting against just 1 false rejection. This often leads to very few or no rejections, as well as an increase the Type II error rate.

A middle-of-the-road approach is to apply the Bonferroni correction to sets of tests that have the same objective. For example, in marginal testing for differential expression of genes, we are looking for a subset of genes which are differentially expressed among healthy and diseased subjects, so it makes sense to do multiple testing correction. On the other hand, it does not make sense to apply a multiple testing correction to all hypothesis tests one has ever done over their entire career.

In S 's situation, even though the 3 hypothesis tests were grouped together in the same paper, they are testing fundamentally different scientific objectives. Hence, a Bonferroni correction across these p -values does not seem appropriate.

Furthermore, it appears that the scientist is conscientious and would report the finding only if $P_{\text{Negative}} > \alpha$, $P_{\text{Confirm}} < \alpha$ and $P_{\text{New}} < \alpha$. However, the announced discovery, only corresponds to the “New” experiment. In particular, from the scientists perspective, the interest is not to announce “New” if it is null. Let us say it is null though, then the probability of making a type-I error is:

$$\mathbb{P}[\text{“New” discovered}] = \mathbb{P}[P_{\text{New}} < \alpha, P_{\text{Confirm}} < \alpha, P_{\text{Negative}} > \alpha] \leq \mathbb{P}[P_{\text{New}} < \alpha] \leq \alpha$$

So indeed it appears that the scientist is more conservative than most by doing these additional checks while running the experiment.

In summary, I would tell S that the reviewers were wrong and suggest that she argue to the reviewers and editors that a Bonferroni correction is not needed in her case because (i) only one new variant of the disease model is being consider (not 3 new ones) and because (ii) the approach of running “Confirm” and “Negative” and making sure they are significant and not significant (respectively) is actually conservative. If S 's paper is still rejected, I would suggest she present the 3 p -values in different sections of the paper (e.g. the “Confirm” and “Negative” can be presented in the methods section while “New” could be presented in the results section), so that future reviewers do not make the same mistake.

Problem 6: A Gaussian Hierarchical Model

Key ideas/tools:

- computations involving convolved Gaussians
- computing the Bayes risk

We assume that the $g_i \mid s$ are conditionally independent of each other.

(a)

$$\begin{aligned} p(s \mid g_1, g_2) &= \frac{p(s, g_1, g_2)}{p(g_1, g_2)} = \frac{p(g_1, g_2 \mid s)p(s)}{\int p(g_1, g_2 \mid s)p(s)ds} \\ &\propto p(g_1 \mid s)p(g_2 \mid s)p(s), \end{aligned}$$

where we have used the assumption on independence of conditional distributions. Using the distributions given in the question,

$$\begin{aligned}
p(g_1 | s)p(g_2 | s)p(s) &\propto \exp \left[-\frac{[g_1 - (s + \Delta)]^2}{2\sigma^2} - \frac{[g_2 - (s - \Delta)]^2}{2\sigma^2} - \frac{s^2}{2} \right] \\
&\propto \exp \left\{ -\frac{1}{2\sigma^2} [(2 + \sigma^2)s^2 - 2(g_1 + g_2)s] \right\} \\
&\propto \exp \left\{ -\frac{2 + \sigma^2}{2\sigma^2} \left[s^2 - 2\frac{g_1 + g_2}{2 + \sigma^2}s \right] \right\} \\
&\propto \exp \left[-\frac{2 + \sigma^2}{2\sigma^2} \left(s - \frac{g_1 + g_2}{2 + \sigma^2} \right)^2 \right].
\end{aligned}$$

Hence, the posterior distribution of $s | g_1, g_2$ is Normal $\left(\frac{g_1 + g_2}{2 + \sigma^2}, \frac{\sigma^2}{2 + \sigma^2} \right)$.

- (b) The fastest way to do this problem is just to immediately notice that the Bayes risk is equal to the expected posterior variance. i.e.

$$\mathbb{E}[(\hat{s} - s)^2] = \mathbb{E}[\mathbb{E}[(\hat{s} - s)^2 | g_1, g_2]] = \mathbb{E}[\text{Var}(s | g_1, g_2)] = \frac{\sigma^2}{2 + \sigma^2}.$$

If you did not notice this fact then you could instead do the brute force computation, which would go as follows. Given s , we have

$$\begin{aligned}
\mathbb{E}_{g_1, g_2} \left[\frac{g_1 + g_2}{2 + \sigma^2} \right] &= \frac{s + \Delta + s - \Delta}{2 + \sigma^2} = \frac{2s}{2 + \sigma^2}, \\
\mathbb{E}_{g_1, g_2} \left[\left(\frac{g_1 + g_2}{2 + \sigma^2} \right)^2 \right] &= \frac{(\sigma^2 + (s + \Delta)^2) + (\sigma^2 + (s - \Delta)^2) + 2(s + \Delta)(s - \Delta)}{(2 + \sigma^2)^2} \\
&= \frac{2\sigma^2 + 4s^2}{(2 + \sigma^2)^2}.
\end{aligned}$$

Thus, the Bayes risk with squared error loss is given by

$$\begin{aligned}
\mathbb{E}_{s, g_1, g_2} [(\mathbb{E}[s | g_1, g_2] - s)^2] &= \int_{-\infty}^{\infty} \left[\frac{2\sigma^2 + 4s^2}{(2 + \sigma^2)^2} - 2s \frac{2s}{2 + \sigma^2} + s^2 \right] \phi(s) ds \\
&= \int_{-\infty}^{\infty} \frac{2\sigma^2 + \sigma^4 s^2}{(2 + \sigma^2)^2} \phi(s) ds \\
&= \frac{2\sigma^2}{(2 + \sigma^2)^2} + \frac{\sigma^4}{(2 + \sigma^2)^2} = \frac{\sigma^2}{2 + \sigma^2}.
\end{aligned}$$

- (c) A fast way to do this question is to notice that $g_3 \stackrel{D}{=} s + z$ where $z \sim N(0, \sigma^2)$ is independent of s , g_1 , and g_2 . We already calculated the distribution of $s | g_1, g_2$

in part a). Moreover, conditional on g_1, g_2 we have that $z \perp s$ with $z \sim N(0, \sigma^2)$. Thus,

$$g_3 \stackrel{D}{=} s+z = \text{Normal}\left(\frac{g_1 + g_2}{2 + \sigma^2}, \frac{\sigma^2}{2 + \sigma^2}\right) + N(0, \sigma^2) = \text{Normal}\left(\frac{g_1 + g_2}{2 + \sigma^2}, \frac{\sigma^2(3 + \sigma^2)}{2 + \sigma^2}\right).$$

Alternatively, one can also derive this result by a direct calculation of the posterior distribution as follows.

$$\begin{aligned} p(g_3 \mid g_1, g_2) &= \int_s p(g_3 \mid s, g_1, g_2) p(s \mid g_1, g_2) ds \\ &= \int_s p(g_3 \mid s) p(s \mid g_1, g_2) ds \\ &\propto \int_{-\infty}^{\infty} \exp\left[-\frac{(g_3 - s)^2}{2\sigma^2} - \frac{2 + \sigma^2}{2\sigma^2} \left(s - \frac{g_1 + g_2}{2 + \sigma^2}\right)^2\right] ds \\ &\propto \int_{-\infty}^{\infty} \exp\left\{-\frac{1}{2\sigma^2} [(g_3 - s)^2 + (2 + \sigma^2)s^2 - 2(g_1 + g_2)s]\right\} ds \\ &= \exp\left(-\frac{g_3^2}{2\sigma^2}\right) \int_{-\infty}^{\infty} \exp\left\{-\frac{1}{2\sigma^2} [(3 + \sigma^2)s^2 - 2(g_1 + g_2 + g_3)s]\right\} ds \\ &\propto \exp\left(-\frac{g_3^2}{2\sigma^2}\right) \exp\left[\frac{3 + \sigma^2}{2\sigma^2} \left(\frac{g_1 + g_2 + g_3}{3 + \sigma^2}\right)^2\right] \\ &\quad \cdot \int_{-\infty}^{\infty} \exp\left[-\frac{3 + \sigma^2}{2\sigma^2} \left(s - \frac{g_1 + g_2 + g_3}{3 + \sigma^2}\right)^2\right] ds \\ &\propto \exp\left[-\frac{g_3^2}{2\sigma^2} + \frac{(g_1 + g_2 + g_3)^2}{2\sigma^2(3 + \sigma^2)}\right] \\ &\propto \exp\left[-\frac{2 + \sigma^2}{2\sigma^2(3 + \sigma^2)} \left[g_3^2 - 2\left(\frac{g_1 + g_2}{2 + \sigma^2}\right) g_3\right]\right]. \end{aligned}$$

Thus, the posterior predictive distribution is $\text{Normal}\left(\frac{g_1 + g_2}{2 + \sigma^2}, \frac{\sigma^2(3 + \sigma^2)}{2 + \sigma^2}\right)$.

- (d) A fast way to do this question is to note that observing g_3 is equivalent to observing $g_1 + \Delta$ and then to apply the result of part a). On the other hand, a

brute force calculation would proceed as follows.

$$\begin{aligned}
p(s \mid g_2, g_3) &\propto p(g_2 \mid s)p(g_3 \mid s)p(s) \\
&\propto \exp \left[-\frac{[g_2 - (s - \Delta)]^2}{2\sigma^2} - \frac{(g_3 - s)^2}{2\sigma^2} - \frac{s^2}{2} \right] \\
&\propto \exp \left\{ -\frac{1}{2\sigma^2} [(2 + \sigma^2)s^2 - 2(g_2 + g_3 + \Delta)s] \right\} \\
&\propto \exp \left[-\frac{2 + \sigma^2}{2\sigma^2} \left(s - \frac{g_2 + g_3 + \Delta}{2 + \sigma^2} \right)^2 \right].
\end{aligned}$$

Hence, the posterior distribution of $s \mid g_2, g_3$ is Normal $\left(\frac{g_2 + g_3 + \Delta}{2 + \sigma^2}, \frac{\sigma^2}{2 + \sigma^2} \right)$.

- (e) Once again the fastest way to get the result is to observe that the Bayes risk is exactly the expected posterior variance. A slower brute force computation would go as follows. Given s , we have

$$\begin{aligned}
\mathbb{E}_{g_2, g_3} \left[\frac{g_2 + g_3 + \Delta}{2 + \sigma^2} \right] &= \frac{(s - \Delta) + s + \Delta}{2 + \sigma^2} = \frac{2s}{2 + \sigma^2}, \\
\mathbb{E}_{g_2, g_3} \left[\left(\frac{g_2 + (g_3 + \Delta)}{2 + \sigma^2} \right)^2 \right] &= \frac{(\sigma^2 + (s - \Delta)^2) + (\sigma^2 + (s + \Delta)^2) + 2(s + \Delta)(s - \Delta)}{(2 + \sigma^2)^2} \\
&= \frac{2\sigma^2 + 4s^2}{(2 + \sigma^2)^2}.
\end{aligned}$$

Thus, the Bayes risk with squared error loss is given by

$$\begin{aligned}
\mathbb{E}_{s, g_2, g_3} [(\mathbb{E}[s \mid g_2, g_3] - s)^2] &= \int_{-\infty}^{\infty} \left[\frac{2\sigma^2 + 4s^2}{(2 + \sigma^2)^2} - 2s \frac{2s}{2 + \sigma^2} + s^2 \right] \phi(s) ds \\
&= \frac{\sigma^2}{2 + \sigma^2},
\end{aligned}$$

which is the same as the Bayes risk we got in (b). This makes sense: since the bias Δ is known, observing g_3 gives exactly the same information as observing $g_3 + \Delta$, which has the same conditional distribution as g_1 .

7 Applied 2018: Solution ⁹

Problem 1: A life table GLM with non-canonical link

Key ideas/tools:

- GLMs with a custom link function

(a) Recall that a GLM consists of the following three elements:

- (1) An exponential family of distributions
- (2) A linear predictor $\eta = X\beta$
- (3) A link function g such that $E(Y | X) = g^{-1}(\eta)$

Our setting fits into this framework, taking the exponential family to be the binomial distribution and the link function g to be the log function. We simply need to specify to R how to fit this model. We can fit it as follows:

```
glm_fit <- glm( (X$Deaths / X$Clients) ~ 1 + X$Age,
               family=binomial(link = "log"),
               weights = X$Clients)
```

(b) We can form the estimate as

$$\prod_{a=35}^{70} [\hat{P}(\text{survive year } a)] = \prod_{a=35}^{70} [1 - e^{\hat{\beta}_0 + \hat{\beta}_1 a}].$$

If any of the estimated probabilities are negative, which is possible with the functional form of our model, we should set them to 0. Note that we can extract the probabilities $e^{\hat{\beta}_0 + \hat{\beta}_1 a}$ manually, or by calling:

```
preds <- predict(glm_fit, type="response")
```

Problem 2: Finding consistent signals

Key ideas/tools:

- Group-Lasso
- Making sure to cross-validate the whole procedure

⁹Stephen Bates, Nikos Ignatiadis, D.K.

We want a model that predicts well at time points $t < 11$. Since there are many features $p > n$ and the signal appears to be weak, we know that a LASSO approach using just Y and X^t does not work well, i.e., it is too difficult to do both feature selection and estimate the coefficients in a way that the final model has good predictive power.

One approach, based on the last bullet of the problem statement, is the following: **(i)** use time points $t \geq 11$ to learn a small “active” set $\mathcal{A} \subset \{1, \dots, p\}$ and then **(ii)** construct models for $t < 11$ by using only the selected features. Let us elaborate on (one approach) of implementing these steps:

Feature selection: We fit a model $Y \sim X^{11:15}$ where $X^{11:15} = [X^{11}, \dots, X^{15}]$. Let us index the coefficient corresponding to X_j^t (protein j at time t) as β_j^t . Note that the j -th column of these matrices corresponds to the same protein. Since we only want to pick a few proteins, we would like to enforce that $\beta_j^{11:15} = 0$ for many j . A method that is great at achieving this is the Group-LASSO, which for a tuning parameter $\lambda > 0$ solves:

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}} \left\{ \|Y - X\beta\|_2^2 + \lambda \sum_{j=1}^{200} \|\beta_j^{11:15}\|_2 \right\}$$

We could pick λ using cross-validation and the 1 standard-error rule to get a more parsimonious model. The returned active sets consists of proteins j such that $\|\beta_j^{11:15}\|_2 > 0$.

Model at time $t < 11$: Fix a $t < 11$. If the feature set \mathcal{A} selected in the first step is parsimonious enough, we could try to even fit a simple linear model, say by OLS of $Y \sim X_{\mathcal{A}}^t$. Then the prediction would be $X_{\mathcal{A}}^t \hat{w}$, where \hat{w} is the $|\mathcal{A}|$ -dimensional coefficient vector estimated above.

An alternative scheme for tuning λ is to run cross-validation, but with objective function equal to the prediction error of the linear model fit using just the predictors from times $t < 11$. This second approach may be more desirable since it is more directly aligns with our goal of fitting a good predictive model for times $t < 11$.

Cross-validation: The last part of the question pertains to cross-validation. Here we need to be careful that there is no information leakage and that we cross-validate the whole procedure [read Chapter 7.10.2 in [Hastie, Tibshirani, and Friedman, 2009]]. For example, we could split the data into 5 folds of 20 observations each. Then, fixing the first fold, we could apply the procedure above based on the remaining 80 observations in the other 4 folds. To tune the Group Lasso we might apply a nested layer of cross-validation (based only on the 80 observations, which we could split into another 5 folds). We then would check how well the whole procedure performs in predicting the first (held-out) fold. We could then repeat the procedure for all 5 folds and average the error.

Problem 3: Hypothesis testing on the genome

Key ideas/tools:

- testing correlations with the t-test
- correcting for multiple comparisons with Bonferroni
- accounting for dependent tests

- (a) A simple first idea is to fit a linear regression between the response variable Y and SNP j given by

$$Y_i = \beta_{0j} + \beta_{1j}X_{ij} + \varepsilon_i.$$

We can use the resulting t-statistic for testing whether or not $\beta_{1j} = 0$ to check whether Y is associated with SNP j . An important thing to note here is that this linear model treats the effect of having multiple copies of the same SNP as additive. This may or may not be biologically plausible (e.g. genes can show dominant/recessive expression where the SNP only matters if you have both copies). An alternative is to model the number of copies of a SNP as a categorical variable with 3-levels and then use an F-test to test if a given SNP is significant. This can equivalently be expressed as a linear model

$$Y_i = \mu_j + \beta_{1j}I\{X_{ij} = 1\} + \beta_{2j}I\{X_{ij} = 2\} + \varepsilon_i$$

where we test the null hypothesis that $\beta_{1j} = \beta_{2j} = 0$ using the F-test for nested linear models.

An even more sophisticated idea to part a) would be to run a multivariate linear regression, including all SNPs in the model (or just some of the SNPs that are thought to be important in the likely scenario that $n < M$). If we used an additive model for the SNPs then we could once again use a t-test to assess the significance of SNP j . In this case, we are testing whether Y is uncorrelated with SNP j , after correcting for the other SNPs. Note that the t-test will not have high power when the variables are highly correlated, like they are in this case. Similar considerations would also apply to F-tests.

- (b) If we carry out the one-by-one tests from the previous part across all M SNPs, then we will still need to adjust the p-values to correct for multiple testing. We can use Bonferroni, multiplying each p-value by 10^6 and considering a SNP significant only if the corrected p-value is below our desired significance threshold α .

We cannot run the full linear model, because the number of variables is larger than the number of observations. Instead, we will have to resort to the one-at-a-time tests that we outlined in the previous part. As a compromise in-between, we could choose try to reduce the number of SNPs in each regression: For example,

we could cluster the SNPs into a manageable number of groups and then run a multivariate linear model with one SNP from each cluster. If we select a SNP, then we can interpret this as evidence that some SNP in the cluster is important.

- (c) The smallest p-value of the M_g SNPs that belong to gene g is not a good measure of gene importance, since it gives an unfair advantage to larger genes that will have more SNPs measured: just by chance, one of these SNPs may have a lower p-value. Of course, this is the same problem as the global null testing problem, so we could avoid it by multiplying the smallest p-value by M_g (i.e., we apply a Bonferroni correction for each gene).

Unfortunately, this is also not optimal: This is “unfair” for the larger genes, since the p-values of nearby SNPs are extremely correlated. One approach used in GWAS is to consider an “effective sample” size that corrects for multiplicity as in Bonferroni but accounting for dependence.

Finally, note that if we want to select multiple genes based on e.g., the per-gene Bonferroni corrected smallest p-value, then we will have the same multiple testing problem described in the previous part. We would again need to correct the p-values, perhaps by another application of Bonferroni (this time with number of tests equal to the number of genes). Since we are scanning across a large number of genes, we will need to see a very low p-value before we can conclude that the effect is real, and so the power may be low.

Problem 4: Covariance estimation with missing data

Key ideas/tools:

- covariance estimation with low-rank data
 - factor models
 - using the blockwise matrix inversion formula
- (a) First of all, we note that it is important to clarify here what the missing data mechanism is. In particular, we assume that the indices of the missing entries should be independent of X .
1. One option is to impute all of the missing entries with the mean or median of that feature, and then compute the empirical covariance matrix.
 2. A second option is to model the features as a multivariate Gaussian, and consider the missing values to be a set of unobserved variables. We can then run the EM algorithm to iteratively updated our guess for the missing values, and then fit the covariance matrix. This will also be positive semi-definite, but the reason why is somewhat subtle. Let $Z \in \mathbb{R}^{n \times p}$ denote the

full augmented dataset with no missing entries. Then, in the M-step we are maximizing

$$\begin{aligned} & \mathbb{E}_{\hat{\mu}^t, \hat{\Sigma}^t} \left[\frac{1}{n} \sum_{i=1}^n (Z_i - \mu)^T \Sigma^{-1} (Z_i - \mu) - \log(|\Sigma|) | X \right] \\ &= \text{Tr}(\Sigma^{-1} \mathbb{E}_{\hat{\mu}^t, \hat{\Sigma}^t} \left[\frac{1}{n} \sum_{i=1}^n (Z_i - \mu)(Z_i - \mu)^T | X \right]) - \log(|\Sigma|). \end{aligned}$$

Now, consider optimizing this last expression over Σ with μ fixed. Since $\mathbb{E}_{\hat{\mu}^t, \hat{\Sigma}^t} \left[\frac{1}{n} \sum_{i=1}^n (Z_i - \mu)(Z_i - \mu)^T | X \right]$ is a PSD matrix we may decompose it as $\mathbb{E}_{\hat{\mu}^t, \hat{\Sigma}^t} \left[\frac{1}{n} \sum_{i=1}^n (Z_i - \mu)(Z_i - \mu)^T | X \right] = \frac{1}{p} V^T V$ for $V \in \mathbb{R}^{p \times p}$. Then, our objective is

$$\text{Tr}(\Sigma^{-1} \frac{1}{p} V^T V) - \log(|\Sigma|) = \frac{1}{p} \sum_{i=1}^p V_i^T \Sigma^{-1} V_i - \log(|\Sigma|).$$

We recognize this last expression (up to additive constants) as the log-likelihood of $N(0, \Sigma)$ with observed data matrix V . We know that this expression is maximized when Σ is the empirical covariance matrix, $\Sigma = \frac{1}{p} V^T V$, which is PSD. Thus, the M-Step will always produce a PSD estimate of the covariance matrix.

3. A third option is to estimate the covariances of X_i and X_j for each pair by taking the empirical covariance after removing any observation that is missing either entry i or entry j . This need not be PSD, although it is symmetric.
4. Let us note that another typical assumption for such problems with large matrices with missing data is to posit that there are indicators $\Xi_{ij} \stackrel{\text{iid}}{\sim} \text{Bernoulli}(\delta)$ for a fixed $\delta \in [0, 1]$. We then only observe X_{ij} if $\Xi_{ij} = 1$. In other words, we could write our observations as:

$$\tilde{X}_{ij} = \Xi_{ij} \cdot X_{ij}$$

Note that this missing-data mechanism makes a stronger assumption than what we assumed for the 3 options above. Let $\tilde{\Sigma}$ be the empirical covariance based on the \tilde{X}_{ij} , then an unbiased estimator of Σ is given by:

$$(\delta^{-1} - \delta^{-2}) \text{diag}(\tilde{\Sigma}) + \delta^{-2} \tilde{\Sigma}.$$

This matrix does not need to be PSD. If we don't know δ , then we can estimate it by the proportion of non-missing entries in X . See Lounici [2014] for this approach.

- (b) If we use any of the approaches (especially options 1 or 3 from part a), then we already have an estimate $\hat{\Sigma}$ of the covariance matrix, that however is not low-rank. We could then compute the projection of $\hat{\Sigma}$ onto PSD matrices of rank K , i.e., let $\hat{\Sigma}_{LR}$ be

$$\hat{\Sigma}_{LR} \in \operatorname{argmin} \left\{ \left\| S - \hat{\Sigma} \right\|_F^2 \mid S \in S_+^p, \operatorname{rank}(S) = 300 \right\}$$

This can be computed as follows: Let $\hat{\Sigma} = \sum \lambda_i v_i v_i^\top$ be the eigenvalue decomposition of $\hat{\Sigma}$. Then let \mathcal{J} be the index set of the 300 largest positive eigenvalues λ_i (if there are less than 300, then \mathcal{J} would contain only the positive eigenvalues). Then:

$$\hat{\Sigma}_{LR} = \sum_{j \in \mathcal{J}} \lambda_j v_j v_j^\top.$$

- (c) The factor analysis model is $\Sigma = LL^\top + \Psi = W + \Psi$ where L is the matrix of latent factors and Ψ is a diagonal matrix with entries ≥ 0 . Given our pilot estimate $\hat{\Sigma}$, we could then let:

$$\hat{\Sigma}_{LRF} \in \operatorname{argmin} \left\{ \left\| S - \hat{\Sigma} \right\|_F^2 \mid S = W + \Psi, W \in S_+^p, \operatorname{rank}(W) = 300, \Psi \geq 0 \text{ diagonal} \right\}$$

It is not clear that we can optimize this to global optimality. However, a heuristic could proceed by alternating minimization. Say we start with $\Psi^0 = 0$ and $W^0 = \hat{\Sigma}_{LR}$ from the previous step. Then we could update $\Psi^1 = \operatorname{diag}(\hat{\Sigma} - W^0)_+$, then reoptimize to get W^1 etc. An alternative, to get a convex formulation, would be to use a nuclear norm constraint instead of the (non-convex) rank constraint.

A further alternative would be to modify the EM algorithm for factor analysis to also account for missing data (as in part a, Option 2).

- (d) We approximate the distribution of the data from a time point as

$$X \sim \mathcal{N}(\hat{\mu}, \hat{\Sigma})$$

where $\hat{\Sigma}$ is the estimate from the previous step, and $\hat{\mu}$ estimated by taking the empirical average of the non-missing entries of each column. Now, at any given time point, about 7% of the data X is unobserved, and so we will reorder the vector so that $X^{(1)} = (X_1, \dots, X_k)$ are the unobserved entries and $X^{(2)} = (X_{k+1}, \dots, X_{475})$ are the observed entries. A good way to impute the missing entries is with the mean of $X^{(1)}$ conditional on the observed entries $X^{(2)}$. Recall that the regression formula (i.e. the formulae for the conditional mean and variance of a multivariate Gaussian) tells us the conditional mean is:

$$\hat{\mu}_{12} := \hat{\mu}^{(1)} + \hat{\Sigma}_{12} \hat{\Sigma}_{22}^{-1} (x^{(2)} - \hat{\mu}^{(2)}).$$

Thus, in order to evaluate the conditional mean efficiently, we will need to compute $\hat{\Sigma}_{12}\hat{\Sigma}_{22}^{-1}(x^{(2)} - \hat{\mu}^{(2)})$ quickly. Note that we cannot just precompute $\hat{\Sigma}_{22}^{-1}$, because the pattern of missingness will be changing with each observation, and hence the matrix $\hat{\Sigma}_{22}^{-1}$ will change. Furthermore, this matrix will have dimension about 440×440 , so it would be slightly costly to invert this matrix each time, although this would certainly take less than a second on a modern computer. In general, if we have d features with 7% missingness, this will take about $C \cdot 0.93^3 \cdot d^3 \approx C \cdot 0.7 \cdot d^3$ operations, where C is some fixed constant.

We can evaluate this even faster than inverting this matrix every time by taking advantage of the blockwise matrix formula, however. First, we compute $\hat{\Sigma}^{-1}$ once explicitly. Then, recall the **blockwise matrix inversion/ Schur complement formula**. In particular

$$\hat{\Sigma}_{22} = ((\hat{\Sigma}^{-1})_{22} - (\hat{\Sigma}^{-1})_{21}(\hat{\Sigma}^{-1})_{11}^{-1}(\hat{\Sigma}^{-1})_{12})^{-1},$$

which implies that,

$$\hat{\Sigma}_{22}^{-1} = (\hat{\Sigma}^{-1})_{22} - (\hat{\Sigma}^{-1})_{21}(\hat{\Sigma}^{-1})_{11}^{-1}(\hat{\Sigma}^{-1})_{12}.$$

Since $(\hat{\Sigma}^{-1})$ is precomputed, we only need to invert the $k \times k$ matrix $(\hat{\Sigma}^{-1})_{11}$ and then do 2 matrix multiplications to do the computation in this way. Since $k \approx 0.07d$, this arrangement of the computation is much faster, requiring approximately $C \cdot k^3 + k^2d \approx C \cdot 0.07^3 \cdot d^3 + 0.07^2 \cdot 0.93 \cdot d^3 \approx C \cdot 0.001 \cdot d^3 + .005 \cdot d^3$ operations (C can be thought of as being larger than 1).

Finally, we note that if the low-rank component from part c) would be of relatively low rank (say much smaller than the dimension), then we could use the Woodbury matrix inversion identity to improve computational efficiency.

Problem 5: Analyzing Gene Expression Data

Key tools/ideas:

- batch effects
 - Gaussian mixture model
 - the EM algorithm
 - non-negative matrix factorization
- (a) We can do PCA and then plot the first two principal components, together with the batch label. If we can see a difference in the distribution of the two classes along the first two PCs, this indicates that we should worry about a batch effect. Depending on what fraction of the variance is explained by the first two PCs we

may also want to look at plots of the other components. We could also make histograms of the empirical means or variances across genes for each of the two batches, and if the histograms differ noticeably, this might indicate that there is a batch effect.

- (b) See Gao, Zhu, Shen, and Pan [2016], Hao, Sun, Liu, and Cheng [2017] for worked out solutions to this problem. We now sketch one approach.

If there are K discrete cell types, we would want to use a **Gaussian mixture model** to describe the data. The model is the following: let $Z_j \sim \text{multinomial}(\pi) \in \{1, \dots, K\}$ be the latent cell type for cell j , and then $X_{.j} \mid Z_j = k \sim \mathcal{N}(\mu_k, \Sigma_k)$. Note that as an alternative, we could force the cell types to have the same covariance matrix $\Sigma_k = \Sigma$, to reduce the number of parameters in the model, but based on the question it sounds like this is not as biologically plausible, so we will allow each class to have its own covariance matrix.

This model can be estimated using the EM algorithm. However, since the dimension is so large with $m > n$, we may want to consider using some sort of penalized covariance estimation within each “M” step. For example we could enforce sparsity of the precision matrices Σ_k^{-1} , as in the Graphical Lasso. The complete-data, penalized log-likelihood, would then take the following form with $\theta = (\pi, \Sigma_1^{-1}, \dots, \Sigma_K^{-1}, \mu_1, \dots, \mu_k)$.

$$\ell(\theta; X; Z) = \sum_{j=1}^n \sum_{k=1}^K \mathbf{1}(Z_j = k) \cdot \log(\pi_k \cdot \phi(X_{.j}; \mu_k, \Sigma_k)) - \lambda \sum_{k=1}^K \|\Sigma_k^{-1}\|_1,$$

where $\phi(x; \mu, \Sigma)$ is the pdf of $\mathcal{N}(\mu, \Sigma)$ evaluated at x .

Now we could iterate the steps of the EM algorithm. The **E**-step is the usual one for mixture models. The **M**-step decomposes into optimization of the proportions π (just average the imputed \hat{Z}_j^t from the E-step), optimization for the μ_k s (weighted averages of the X s) and finally K optimizations for the Σ_k^{-1} , each of which is a weighted Gaussian Graphical Lasso problem.

To choose K and λ , we could use *BIC* or a cross-validation scheme where we predict the gene expression levels of a subset of genes base on the remaining genes on a hold-out set of observations.

To decide whether or not to transform the data, we could try to find a sample of cells where we are confident that they have the same cell type. We can then check the marginal distribution to see if they are approximately Gaussian (by testing skewness and kurtosis), and if not, we can use the appropriate transformation. Since the features are counts, one plausible generating distributions (within each cluster) is a Poisson distribution, in which case the data might be skewed. The **Anscombe transform**, $2\sqrt{x + 3/8}$, transforms Poisson variables with large

mean λ into variables with distribution approximately $\mathcal{N}(2\sqrt{\lambda + 3/8}, 1)$, so this would be a natural transformation to consider in this setting. A negative Binomial model would perhaps fit even better; in which case one could use a variance stabilizing transformation for the negative binomial model (which would require a preliminary estimate of the dispersion, cf. Anders and Huber [2010]).

Alternative solution: If the genes are highly correlated, a sparse precision matrix may be an unpalatable assumption. If the genes are highly correlated, a simple approach would be to fit a standard Gaussian mixture model on the first n_p (say 50) principle component scores. K and n_p could be chosen by AIC or BIC in this case. One potential issue with this approach is “double dipping”. Fitting PCA and the Gaussian on the same dataset may not be appropriate. You can use sample splitting to assure you avoid this issue. In particular you could fit PCA on the random selection of $n/5 = 2000$ of the samples. This will tell you the principal component directions (or the weights for linear combinations of the data that explain the most variance). Then on the remaining 4/5ths of the data, you can apply the linear combinations describing the leading n_p principal component directions to the raw data to produce n_p features on the remaining data. These n_p features will also follow a Gaussian mixture model, because a linear map applied to a multivariate Gaussian is a multivariate Gaussian and because we assume the raw data follows a Gaussian mixture model.

- (c) The matrix A can be interpreted as k different prototype gene expression profiles. Then, the matrix AB represents each cell as a positive mixture of the k prototypes, with the non-negative weights for each cell given in the B matrix.

Problem 6: Privacy-preserving data releases

Key ideas/tools:

- Differential privacy
- interpreting total variation distance and likelihood ratio bounds

Note in the question there is a comment that ϵ must satisfy $0 < \epsilon < 1$ to rule out trivial cases. This is only true of ϵ -TV-privacy. For ϵ -LR-privacy, all $\epsilon > 0$ can be considered.

- (a) Suppose we wish to test that an individual's data is x . Then any test with type-I error bounded by α can have power at most $\alpha + \epsilon$. An equivalent interpretation is that for any hypothesis test of $H_0 : X = x$ versus $H_1 : X = x'$, the type I error plus the type II error must be at least $1 - \epsilon$. To see this let A be the event that the hypothesis test rejects the null, then under ϵ -TV privacy

$$\epsilon \geq |P_0(A) - P_1(A)| = |\text{Type-I-error} + (1 - \text{Type-II-error})|$$

and therefore,

$$\text{Type-I-error} + \text{Type-II-error} \geq 1 - \epsilon.$$

- (b) The second definition is stronger, at least for small ϵ . It implies that

$$Q(Z \in A|X = x) - Q(Z \in A|X = x') \leq e^\epsilon - 1,$$

and so exchanging x and x' , we arrive at

$$|Q(Z \in A|X = x) - Q(Z \in A|X = x')| < e^\epsilon - 1,$$

for any A . Therefore, ϵ -LR-privacy implies ϵ' -TV-privacy for some $\epsilon' > 0$, namely $\epsilon' = e^\epsilon - 1$.

However, ϵ -TV-privacy does not necessarily imply any level of ϵ' -LR-privacy. As an example, suppose $K = 1$ and let $Q(\cdot | X = x)$ be given by,

$$\begin{aligned} Q(Z = 1|X = 0) &= 1, \\ Q(Z = 1|X = 1) &= 1 - \epsilon, \\ Q(Z = \perp |X = 1) &= \epsilon. \end{aligned}$$

That is, when $X = 0$, we always return $Z = 1$ but when $X = 1$ we return $Z = 1$ with probability $1 - \epsilon$ and return \perp otherwise. Note that the TV between $Q(\cdot|X = 0)$ and $Q(\cdot|X = 1)$ is exactly ϵ . However, the likelihood ratio $Q(Z = \perp |X = 1)/Q(Z = \perp |X = 0)$ is infinite and cannot be bounded by $e^{\epsilon'}$ for any $\epsilon' > 0$. In this example, if \perp is observed, we know for sure $X = 1$, which could be a privacy concern.

- (c) We can return $Z = \perp$ with probability $1 - \epsilon$, and the value of X otherwise. Then, we can take the empirical mean of values of Z that are not \perp . This satisfies the TV definition of privacy, since the responses agree with probability $1 - \epsilon$, no matter what the value of X is.

Let N be the number of observations that are not \perp . Conditional on $N > 1$, the estimator is unbiased, with variance $\frac{\theta(1-\theta)}{N}$. Since $\frac{N}{n} \rightarrow \epsilon$ almost surely as $n \rightarrow \infty$, asymptotically we have that the MSE is $\frac{\theta(1-\theta)}{n\epsilon}$. This estimator is clearly \sqrt{n} consistent, and has MSE that goes to 0 as $\theta \rightarrow 0$ or $\theta \rightarrow 1$, provided $N \geq 1$.

- (d) With probability β , we return 0 or 1 uniformly at random, otherwise we return the value of X . With this scheme, we have

$$\beta/2 \leq Q(Z = z|X = x) \leq 1 - \beta/2,$$

for all z and x . The likelihood ratio thus satisfies,

$$\frac{Q(Z = z|X = x)}{Q(Z = z|X = x')} \leq \frac{1 - \beta/2}{\beta/2} = \frac{2 - \beta}{\beta} = e^\epsilon,$$

where $\epsilon = \log(2 - \beta) - \log(\beta)$. The procedure $Q(\cdot|X = x)$ is thus e^ϵ -LR-private. We then take the estimator

$$\hat{\theta} = \frac{1}{1 - \beta}(\bar{Z} - \beta/2).$$

Note this is unbiased with variance

$$\frac{1}{(1 - \beta)^2 n}((1 - \beta)\theta + \beta/2)(1 - (1 - \beta)\theta - \beta/2),$$

so this is the MSE which show \sqrt{n} consistency.

8 Applied 2019: Solution¹⁰

Problem 1: Analyzing a randomized clinical trial

Key ideas/tools:

- log-rank (Mantel–Haenszel) test
- Cox proportional hazards model

Our dataset includes 40 patients, some of which were treated with a new drug ($W_i = 1$) and some of which received the current standard of care ($W_i = 0$). For each patient we record covariates X_i that include sex, age and BMI. Furthermore, we record two dates: when a patient was first treated (F_i) and when they died (D_i). The trial ends on December 31, 2014.

We now set this problem up in survival analysis notation: Let $\delta_i \in \{0, 1\}$ be an indicator of whether the person died before December 31, 2014. In that case we write $\delta_i = 1$ and define the (observed) survival time $O_i = T_i = D_i - F_i$ (with the difference defined in terms of days). If $\delta_i = 0$, we only know that the (unobserved) survival time T_i satisfies $T_i \geq O_i = \text{December 31, 2014} - F_i$.

After this minor preprocessing, our data consists of $(O_i, \delta_i, W_i, X_i)$, $i = 1, \dots, 40$ and we seek to test whether the new drug increased the survival time. Let us further arrange this data in a `data.frame` with columns:

Time, Status, Treatment, Age, Sex, BMI.

A very standard analysis here to test the null hypothesis of no treatment effect (i.e., equal survival curves for treated and untreated) is the log-rank test, which can be called in ‘R’ as follows

```
library(survival)
survdif(Surv(Time, Status) ~ Treatment, data=data)
```

The result above does not directly give an answer about directionality. However, one can check directionality by checking whether `Observed - Expected` is < 0 for the treated group (and standard modifications to implement a directional test).

There are a few aspects worth mentioning here: under the randomization, the log-rank test is valid (as long as we have *i.i.d.* samples) without further assumptions. However, it is most powerful under alternatives that satisfy the proportional hazards assumption. And indeed, the log-rank test is the same as the Score test in the following Cox proportional hazards model:

```
coxph_fit <- coxph(Surv(Time, Status) ~ Treatment, data=data)
summary(coxph_fit)
```

¹⁰Nikos Ignatiadis and Dan Kluger

Note that so far we have ignored the covariates X_i ; and indeed this is commonly done in the analysis of randomized trials. With the Cox proportional hazards model, we could however include the covariates:

```
coxph_cov_fit <- coxph(Surv(Time, Status) ~ Treatment + Age +
  Sex + BMI, data=data)
summary(coxph_cov_fit)
```

In this case, instead of inspecting the score-test p-value, we would inspect the p-value/confidence interval corresponding to the coefficient of W_i (treatment); a negative coefficient provides evidence that the survival time is longer with the new drug. Note that in the above model, we assume that the hazard rate at time t of an individual with treatment status w_i and covariates x_i is equal to

$$h(t; w, x) = h_0(t) \cdot \exp(\alpha \cdot w + \beta^\top x),$$

where $h_0(\cdot)$ is an unspecified baseline hazard function. The interpretation of coefficients needs to take account of this model.

Note that you would likely get full credit on the exam if you simply described the Cox Proportional hazards model with covariates and the R call, without discussing the log-rank test. Although it cannot hurt to mention the log-rank test (or the equivalent score test) as an option if we are willing to ignore Age, Sex, and BMI covariates.

Problem 2: Multivariate analysis with censored Gaussian measurements

Key ideas/tools:

- EM algorithm for Censoring
 - Factor analysis
 - See Augugliaro, Abbruzzo, and Vinciotti [2020] and references therein for recent research in the spirit of this question.
- (a) The data-generating mechanism in this problem is $X_i \sim \mathcal{N}(0, \Sigma)$, $i = 1, \dots, n$. However, we instead observe censored observations X_i^* . The statement is slightly ambiguous about the censoring mechanism: my interpretation is that we only observe entries X_{ij} if $X_{ij} \leq c$, otherwise an error is returned¹¹. Let $\delta_{ij} = \mathbf{1}(X_{ij} > c)$ and $X_{ij}^* = X_{ij}$ if $\delta_{ij} = 1$ or equal to an arbitrary value otherwise (we take that value to be c). The data available to us is $\mathbf{X}^* = (X_{ij}^*)_{ij}$ and $\boldsymbol{\delta} = (\delta_{ij})_{ij}$.

¹¹The alternative interpretation is that if any entry X_{ij} is $> c$, then we do not get to see the whole vector X_i .

Let us first write down the full-data – i.e., with the full data matrix $\mathbf{X} = (X_{ij})_{ij}$ observed – log-likelihood (up to an additive constant): letting $S = \mathbf{X}^\top \mathbf{X} / n$ the empirical covariance matrix, it holds that

$$-\ell(\Sigma; \mathbf{X}) = \frac{n}{2} \text{trace}(S \Sigma^{-1}) + \frac{n}{2} \log(|\Sigma|) + \frac{np}{2} \log(2\pi).$$

Note that the negative log-likelihood above is convex in terms of the inverse precision matrix $\Theta = \Sigma^{-1}$.

The EM algorithm here would proceed as follows. We could initialize the algorithm for example with $\hat{\Sigma}^0 = (\mathbf{X}^*)^\top \mathbf{X}^* / n$ (which should be full rank). Then, let's say that in the t -th iteration our estimate is $\hat{\Sigma}^t$. The E-M steps then proceed as follows:

E-step: We seek to compute:

$$-\tilde{\ell}^{t+1}(\Sigma) = \mathbb{E}_{\hat{\Sigma}^t} [-\ell(\Sigma; \mathbf{X}) \mid \mathbf{X}^*, \delta].$$

By linearity, it thus suffices to compute

$$\hat{S}^{t+1} = \mathbb{E}_{\hat{\Sigma}^t} [S \mid \mathbf{X}^*, \delta]$$

M-step: We get that $\hat{\Sigma}^{t+1} = \hat{S}^{t+1} \in \text{argmax}_{\Sigma} \tilde{\ell}^{t+1}(\Sigma)$. It may not be obvious why this is the case. One way to see this is to note that $\hat{S}^{t+1} = \frac{1}{p} V V^\top$ for some matrix V and so the expected log-likelihood is exactly the observed data log-likelihood for data $V_1, \dots, V_p \sim N(0, \Sigma)$. We know that in this case $\hat{\Sigma}^{t+1}$ is exactly the empirical covariance of V_1, \dots, V_p , i.e. it is \hat{S}^{t+1} . Another way to show this is to observe that by properties of the determinant and linearity of trace, the expected negative log-likelihood equals

$$-\tilde{\ell}^{t+1}(\Sigma) = \frac{n}{2} \text{trace}(\hat{S}^{t+1} \Sigma^{-1}) - \frac{n}{2} \log(|\Sigma^{-1}|) + \frac{np}{2} \log(2\pi),$$

so by formulas (57) and (100) in the matrix cookbook the gradient with respect to the precision matrix $\Theta = \Sigma^{-1}$ is given by

$$\nabla_{\Theta} [-\tilde{\ell}^{t+1}(\Sigma)] = \frac{n}{2} (\hat{S}^{t+1})^T - \frac{n}{2} \Sigma,$$

and setting the gradient to zero yields $\hat{\Sigma}^{t+1} = \hat{S}^{t+1}$ maximizes $\tilde{\ell}^{t+1}(\Sigma)$.

- (b) As seen above, the M -step does have a closed form solution, while the E -step does not (and is also not trivial to implement). Let us describe how we could tackle the implementation of the E-step.

First, note that by decomposing

$$S = \frac{1}{n} \sum_{i=1}^n X_i X_i^\top,$$

we see that it suffices to compute for all $i = 1, \dots, n$ and $k, \ell = 1, \dots, p$

$$\mathbb{E}_{\widehat{\Sigma}^t} [X_{ik}X_{i\ell} \mid \mathbf{X}^*, \boldsymbol{\delta}] = \mathbb{E}_{\widehat{\Sigma}^t} [X_{ik}X_{i\ell} \mid X_i^*, \delta_i].$$

Let us distinguish three cases:

- If both $X_{ik}, X_{i\ell} < c$, then no imputation is necessary, i.e., $\mathbb{E}_{\widehat{\Sigma}^t} [X_{ik}X_{i\ell} \mid X_i^*, \delta_i] = X_{ik}^*X_{i\ell}^*$.
- If $X_{ik} > c$, but $X_{i\ell} \leq c$ (analogously also with k and ℓ flipped), then $\mathbb{E}_{\widehat{\Sigma}^t} [X_{ik}X_{i\ell} \mid X_i^*, \delta_i] = X_{i\ell} \mathbb{E} [X_{ik} \mid X_i^*, \delta_i]$ and so we need to compute $\mathbb{E}_{\widehat{\Sigma}^t} [X_{ik} \mid X_i^*, \delta_i]$ when $\delta_{ik} = 1$. The tractability of this step will in general depend on $\#\{j : \delta_{ij} = 1\}$. For example, if $\delta_{ij} = 0$ for all $j \neq k$, then:

$$\mathbb{E}_{\widehat{\Sigma}^t} [X_{ik} \mid X_i^*, \delta_i] = \mathbb{E}_{\widehat{\Sigma}^t} [X_{ik} \mid X_{ik} > c, X_{i(-k)}],$$

where $X_{i(-k)} = (X_{ij})_{j \neq k}$. However, we know that for $X_i \sim \mathcal{N}(0, \widehat{\Sigma}^t)$, it holds that:

$$X_i \mid X_{i(-k)} \sim \mathcal{N}(\tilde{\mu}, \tilde{\sigma}^2),$$

with

$$\tilde{\mu} = \widehat{\Sigma}_{k,(-k)}^t \left(\widehat{\Sigma}_{(-k),(-k)}^t \right)^{-1} X_{i(-k)}$$

and

$$\tilde{\sigma}^2 = \widehat{\sigma}_k^2 - \widehat{\Sigma}_{k,(-k)}^t \left(\widehat{\Sigma}_{(-k),(-k)}^t \right)^{-1} \widehat{\Sigma}_{(-k),k}^t$$

Then, by standard calculations for truncated Normal distributions, we would get

$$\mathbb{E}_{\widehat{\Sigma}^t} [X_{ik} \mid X_{ik} > c, X_{i(-k)}] = \mathbb{E}_{\mathcal{N}(\tilde{\mu}, \tilde{\sigma}^2)} [X_{ik} \mid X_{ik} > c] = \tilde{\mu} + \tilde{\sigma} \varphi(c) / (1 - \Phi(c)),$$

with φ, Φ , the pdf and CDF of the standard Normal distribution.

Note that if e.g., $\#\{j : \delta_{ij} = 1\} = d$, then we could condition on the $p - d$ non-censored variables and then we would need to integrate with respect to a d -dimensional Gaussian distribution with all coordinates truncated to be $> c$. This is still tractable as long as d remains small; see Tallis [1961] for an elegant way of doing this.

- If both $X_{ik}, X_{i\ell} > \delta$, then we need to compute $\mathbb{E}_{\widehat{\Sigma}^t} [X_{ik}X_{i\ell} \mid X_i^*, \delta_i]$. For example, in case $k = \ell$ and $\#\{j : \delta_{ij} = 1\} = 1$, then we would need to compute $\mathbb{E}_{\widehat{\Sigma}^t} [X_{ik}^2 \mid X_{i(-k)}, X_{ik} > c]$, which we could achieve by computing the 2nd moment of the truncated Gaussian variable we studied above. Other cases are more complicated, although if there are not too many variables censored again we can use the method of Tallis [1961].

Perhaps an easier alternative to computing the integrals exactly would be to use the MCEM algorithm (Monte Carlo EM). The idea is as follows: At iteration t , and for $b = 1, \dots, B_t$ (where B_t will typically become larger as iterations increase), we draw samples:

$$X_i^b \sim \left[\mathcal{N} \left(0, \hat{\Sigma}^t \right) \mid X_i^*, \delta_i^* \right], \quad i = 1, \dots, n$$

Here \sim could be implemented either exactly (say, by rejection sampling), or perhaps more realistically by a Monte Carlo algorithm (e.g., a Gibbs sampler seems to be tractable here).

Then we would replace $\mathbb{E}_{\hat{\Sigma}^t} [S \mid \mathbf{X}^*, \boldsymbol{\delta}]$ in the E -step by its sample average approximation, i.e.,

$$\frac{1}{n \cdot B_t} \sum_{i=1}^n \sum_{b=1}^{B_t} X_i^b (X_i^b)^\top$$

Note that the estimated covariance matrix $\hat{\Sigma}$ will be PSD as long as \hat{S}^t is PSD throughout all steps, which it should be.

- (c) The EM algorithm above will produce an estimate $\hat{\Sigma}$ of the population covariance matrix Σ . To perform PCA, we could form the eigendecomposition $\hat{\Sigma} = V \Gamma V^\top$. The columns of V then would define the principal component directions. Furthermore, the variance along the j -th eigenvalue direction is $\text{Var} [X_i^\top V_j] = \Gamma_{jj}$. (Computing the principal components on the original dataset, \mathbf{X} , however is not directly feasible when \mathbf{X} is not fully observed, although at least the principal component scores for observations X_i^* with no errors can be computed with $(X_i^*)^\top V$. One approach to obtain principal component scores for all n samples would be to impute \mathbf{X} based on the last step of the EM algorithm above and then compute the principal components with $\mathbf{X}_{\text{imputed}} V$.)
- (d) In the factor model, we model the covariance matrix as

$$\Sigma = LL^\top + \Psi,$$

where Ψ is a diagonal matrix and L is a $p \times k$ matrix for $k < p$. Such a covariance structure is implied by the following latent variable model:

$$Y \sim \mathcal{N}(0, I_k), \quad X \mid Y \sim \mathcal{N}(LY, \Psi)$$

Fitting this factor model, when all X_i are fully observed, typically proceeds by the EM algorithm based on the complete-data log-likelihood $\ell(L, \Psi; \mathbf{X}, \mathbf{Y})$ where \mathbf{Y} is the matrix with rows equal to the latent Y_i^\top . The E -step consists of computing the expectations of $Y_i X_i^\top$ and $Y_i Y_i^\top$ conditionally on X_i . The M -step has a closed form solution.

In the setting of this problem, we can keep the same M-step but need to modify the E-step: concretely, we would need to compute the expectations of $Y_i X_i^\top$, $Y_i Y_i^\top$ and $X_i X_i^\top$ conditionally on X_i^*, δ_i given our current guesses for \hat{L}^t and $\hat{\Psi}^t$.

Problem 3: Diaconis and Ylvisaker [1985] priors

Key ideas/tools:

- Conjugate priors in Bayesian inference.
 - More flexible priors by mixing of conjugate priors.
- (a) Throughout this problem we assume that $p_k(\theta) \in \mathcal{Q}$ for $k = 1, \dots, K$, where \mathcal{Q} is a class of conjugate prior densities for the sampling model $y \mid \theta$. This means that for any $q \in \mathcal{Q}$ it also holds that $q(\theta \mid y) \in \mathcal{Q}$. We next seek to study the (larger) class of prior densities of the form:

$$\mathcal{P} = \left\{ p(\theta; \lambda) = \sum_{k=1}^K \lambda_k p_k(\theta), \lambda_k \geq 0, \sum_{k=1}^K \lambda_k = 1, p_k \in \mathcal{Q}, k = 1, \dots, K \right\} \quad (36)$$

Fix a $p \in \mathcal{P}$, which is determined by p_1, \dots, p_K and $\lambda_1, \dots, \lambda_K$. Let us derive the posterior and show that $p(\cdot \mid y) \in \mathcal{P}$ as well (so that \mathcal{P} is indeed closed under sampling). To this end, it is convenient to introduce the latent variable Z_i such that $\mathbb{P}[Z_i = k] = \lambda_k$. Then we can generate $\theta \sim p$ by sampling $\theta \mid Z = k \sim p_k$. The posterior may be decomposed as:

$$p(\theta \mid y) = \sum_{k=1}^K p(\theta \mid y, Z = k) \mathbb{P}[Z = k \mid y] = \sum_{k=1}^K p_k(\theta \mid y) \mathbb{P}[Z = k \mid y]$$

Now note that $p_k(\cdot \mid y) \in \mathcal{Q}$ by conjugacy of the p_k and we may identify the vector $(\mathbb{P}[Z = 1 \mid y], \dots, \mathbb{P}[Z = K \mid y])$ with an element $\tilde{\lambda}$ on the probability simplex. So indeed:

$$p(\cdot \mid y) \in \mathcal{P},$$

i.e., \mathcal{P} is closed under sampling. To complete the answer, let us also derive the explicit form of $\tilde{\lambda}_k = \mathbb{P}[Z = k \mid y]$. We have:

$$\tilde{\lambda}_k = \mathbb{P}[Z = k \mid y] = \frac{p(y \mid Z = k) \mathbb{P}[Z = k]}{p(y)} = \frac{\lambda_k \cdot m_k(y)}{\sum_j \lambda_j m_j(y)},$$

where $m_k(y) = \int p(y \mid \theta) p_k(\theta) d\theta$.

- (b) Let $\pi \in (0, 1)$ be a small fixed constant. We can model the location parameter θ as coming from the following mixture:

$$\theta \sim (1 - \pi)\mathcal{N}(1, 0.5^2) + \pi\mathcal{N}(-1, 0.5^2) \quad (37)$$

This prior is precisely of the form (36) where the sampling model is that of Normal location family, i.e., for a fixed $\sigma^2 > 0$:

$$y \mid \theta \sim \mathcal{N}(\theta, \sigma^2),$$

and with $\mathcal{Q} = \{\mathcal{N}(\mu, \tau^2) : \mu \in \mathbb{R}, \tau^2 > 0\}$. Note that since we think that θ is much more likely to be close to 1 than -1 , we would pick π to be close to 0, for example, $\pi = 0.05$.

- (c) Now we observe y_1, \dots, y_{10} i.i.d. from the above sampling model with $\sigma^2 = 1$. Furthermore, it happens that $\bar{y} = -0.25$. We seek to derive the posterior w.r.t. the prior (37).

First let us note that by sufficiency, we may collapse the data y_1, \dots, y_{10} , to \bar{y} , with the sampling model

$$\bar{y} \mid \theta \sim \mathcal{N}(\theta, 1/10)$$

To compute the posterior distribution we need to compute $p_k(\theta \mid \bar{y})$ and $m_k(y)$. For $m_k(y)$ by standard convolution arguments of two Gaussians, we get that $m_k(y)$ is the density of $\mathcal{N}(\mu_k, 1/10 + 1/4) = \mathcal{N}(\mu_k, 7/20)$ with $\mu_1 = 1$ and $\mu_2 = -1$ and so introducing latent Z as in part a), we get the updated mixing probability

$$\tilde{\pi} = \mathbb{P}[Z = 2 \mid y] = \frac{\pi \exp(-0.75^2 \cdot 10/7)}{\pi \exp(-0.75^2 \cdot 10/7) + (1 - \pi) \exp(-1.25^2 \cdot 10/7)}$$

On the other hand, let us derive $p_k(\theta \mid y)$. By conjugacy, these are normal with posterior mean

$$\tilde{\mu}_k = \mathbb{E}_k[\theta \mid \bar{y}] = \frac{\bar{y}/4 + \mu_k/10}{1/10 + 1/4} = \frac{10\bar{y} + 4\mu_k}{14} \implies \tilde{\mu}_1 = 1.5/14, \tilde{\mu}_2 = -6.5/14$$

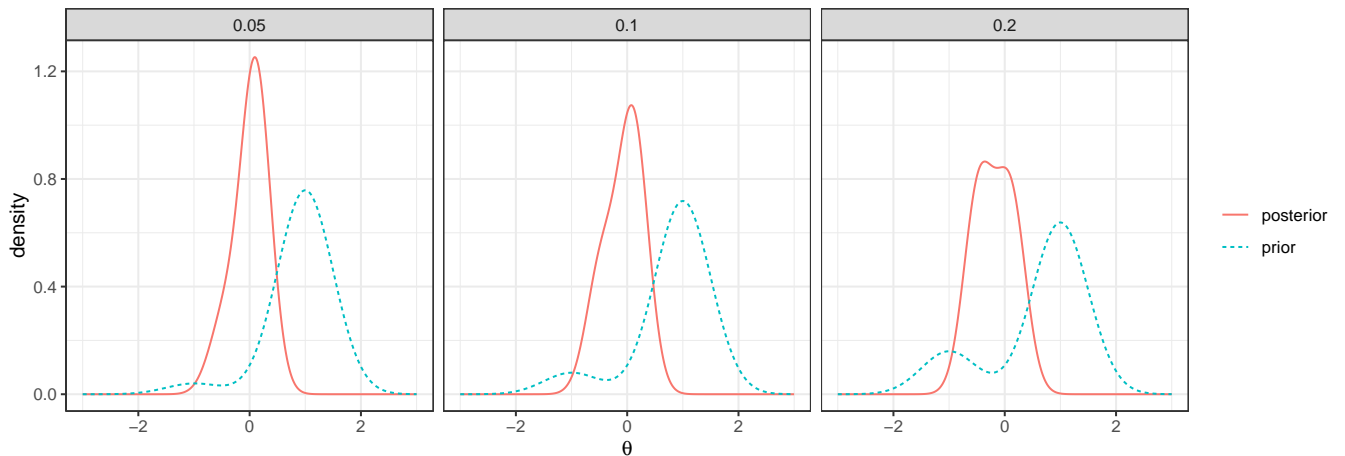
and per-component posterior precision equal to:

$$\text{Var}_k[\theta \mid \bar{y}]^{-1} = 0.1^{-1} + 0.25^{-1} = 14$$

Putting everything together,

$$\theta \mid \bar{y} \sim (1 - \tilde{\pi})\mathcal{N}(1.5/14, 1/14) + \tilde{\pi}\mathcal{N}(-6.5/14, 1/14) \quad (38)$$

- (d) Let us create .. exact .. plots of the prior and posterior for 3 different values of π , namely $\pi \in \{0.05, 0.1, 0.2\}$. The code used is displayed below the plot.



```
library(tidyverse)

y_bar = -0.25

post_mean_1 <- (0.25*y_bar + 0.1*1)/0.35
post_mean_2 <- (0.25*y_bar + 0.1*(-1))/0.35
post_std <- sqrt(1/14)

posterior_pi <- function(pi) {
  pi*dnorm(y_bar,-1.0,sqrt(0.35))/((1-pi)*dnorm(y_bar,
    1.0,sqrt(0.35)) + pi*dnorm(y_bar,-1.0,sqrt(0.35)
  ))) }

plot_df_fun <- function(pi){
  data.frame(theta = seq(-3,3, length=1000)) %>%
    mutate(prior = (1-pi)*dnorm(theta, 1, 0.5) + pi*
      dnorm(theta, -1, 0.5),
      posterior_pi = posterior_pi(pi),
      posterior = (1-posterior_pi)*dnorm(theta,
        post_mean_1, post_std) +
        posterior_pi*dnorm(theta,
          post_mean_2, post_std
        ),
      pi=pi)}

plot_df <- bind_rows(lapply(c(0.2, 0.1, 0.05), plot_df_fun))

plot_df <- pivot_longer(plot_df, -c(theta,pi,posterior_pi),
  names_to="distribution", values_to="density")
```

```
ggplot(plot_df, aes(x=theta, y=density, col=distribution,
  linetype=distribution)) +
  geom_line() +
  xlab(expression(theta)) +
  theme_bw() +
  facet_grid(.~pi) +
  theme(legend.title=element_blank())
```

- (e) This problem asks for an explicit expression of the mode of $\theta \mid \bar{y}$. It is just the argmax of the density in (38), which is easy to compute numerically. I do not think there is an analytic solution to the mode; indeed as per the figure above we see that the posterior could be bimodal as well. Here are a few things we can say (which we can verify e.g., by taking the derivative of the objective): the mode(s) must lie in $(\tilde{\mu}_2, \tilde{\mu}_1) = (-6.5/14, 1.5/14)$. Furthermore, as $\pi \rightarrow 0$, the mode converges to 1.5/14 (I believe the answer the question writer was going for was 1.5/14 and they meant to ask for an approximation of the posterior mode when π is small). If you enjoy these kinds of calculations, also see Behboodian [1970].

Problem 4: Assessing a clinical trial's conclusion

Key ideas/tools:

- Interaction effects in linear models.
- Intent-to-treat analyses.
- p-value interpretation.

- (a) Let us describe our main points of disagreement/agreement with the study:

Patients that dropped out: A major issue with this study are the 51 patients that dropped out. If they had not dropped out, we would have a randomized experiment, and could even draw causal conclusions out of the two-sample comparison. Now, instead, we need to assume that the dropout of these patients was independent of their counterfactual (i.e., the response we would have observed, had they remained in their study). This may not be true, e.g., perhaps for some patients the drug has an adverse effect, and actually increases their blood glucose level. If these are these patients that dropped out, then the causal conclusions of the study will be flawed.

These situations arise all the time in medical studies, e.g., indeed, if a medication has side-effects on a patient, then it is not reasonable to expect them to keep taking that medication. However, there is a well-known remedy used: the

investigators could have measured the glucose level at the end of the trial also for the patients that stopped taking the medication. Then we would use the measured response for these patients in an analysis, pooling them together with other treated patients. Such an analysis would produce an unbiased estimate of the treatment effect that however accounts for the (lack of) adherence of patients to the medication. This is called an **Intention-To-Treat (ITT)** effect.

Unfortunately, here it seems the investigators did not/could not collect such intention-to-treat data. I would do a few sanity checks here (and perhaps report them as a sensitivity analysis in the manuscript's supplement): one check would be to conservatively "impute" a value for the patients that dropped out, say with 0 (no change in glucose level), and rerun all analyses. If conclusions hold up even under a "conservative", then we would be more confident in them. Another potential check would be to check if dropout is correlated with other patient-level variables (in case such data is available).

Reported effect and p-value: The authors write that "This study shows that the drug reduces glucose levels in subjects with moderately high glucose levels by more than placebo (t-test, $p < 0.0001$)": It seems that this sentence refers to the two sample t-test that had a p-value of 0.0004 which is not < 0.0001 . Nevertheless, I mostly agree with this statement: modulo the patient drop-out considerations above, both the regression analysis and the two sample t-test support the conclusion that the drug reduced glucose level.

Interaction effects: The authors further write that "[the regression analysis] reveals that the drug is more effective in males than in females": The coefficient for 'sex' is not correctly interpreted here. The coefficient for 'sex' hints that the decrease in glucose was stronger for males than females; across *all* study subjects (i.e., whether treated or not). However it does not say anything about the relative effectiveness of the drug according to sex. The same erroneous interpretation is also applied to all other coefficients. Instead, one could try to answer the type of question asked here by including interaction effects in the regression, for example, `treatment * sex`.

Study population: The authors write "we recommend that this drug be prescribed to every patient who wants to lower his/her glucose level." Critically, the study did not include diabetic patients who have high glucose levels and may be the most interested in taking the drug. Thus, recommending the drug to all potential patients seem premature without further study in this group. Even within the study population, the authors have not sufficiently backed up this claim. For example, they haven't shown evidence that the drug should be prescribed to both males and females, but they could easily add this evidence by conducting two

t-tests, or fitting their regression model with an interaction term `treatment * sex`, and checking that the magnitude of the interaction term is smaller than the magnitude of the overall treatment effect.

- (b) Let us first outline concerns with the new study. In short, I do not believe the new study contradicts the first study.

p-values and power: The major problem with this new study is that it misinterprets the p-value, i.e., it conflates the absence of evidence with the evidence of absence. While a small p-value provides evidence that we can reject the null hypothesis, a large p-value is uninformative: the p-value could be large either because the null is true or because the study is not powered sufficiently. Indeed; we only have a 12 vs. 12 comparison; a much smaller sample size than the first study. Furthermore, note that the p-value is still somewhat small and the authors of the new study commit the aforementioned fallacy and furthermore over-interpret the arbitrary cutoff of 0.05.

Generalizability: The study recruitment criteria are slightly different than in the original study (e.g., start glucose levels 95-110 vs 100-120 in the original study, similarly for age) and so the populations of the two studies are also slightly different. However, I would not be worried too much about this here.

On the other hand, up to the small sample size, this appears like a nice randomized study that furthermore does not suffer from the dropout of the original study. I would thus start by looking at the 95% confidence interval for $\mu_x - \mu_y$ in the new study. Even though we know that it will cross 0, it would be promising to compare it to the confidence interval of the original study. Indeed, I think it is likely that the new confidence interval could provide further evidence for the effect found in the first study. One could also try to combine the two treatment effect estimates through a small meta-analysis (or if raw-data is available, use both studies in the same linear regression).

Problem 5: Two ways of bootstrapping

Key ideas/tools:

- Time-series bootstrap.
- Bootstrapping individuals.

Statistician A's procedure does not make sense to me; there are some details missing from the description. Thus I think that this problem is quite open-ended and that the answer will depend on the interpretation of what Statistician A actually intends to do. Just make sure to describe in detail how you interpreted Statistician A, before

providing an answer! In this solution we will assume that A's suggestion is to re-sample swimmer's with replacement.

One way to conceptualize what is going on, is as follows: there are two sources of variation in this dataset. First, swimmers are different from each other. Second, the time series measurements (i.e., the speed of the swimmers on 40 consecutive days) are also random; even when we only look at a single swimmer: each time they swim, there's a random source of variation determining how long it will take them to finish the 100 meters; and furthermore there is correlation across days: there could be hot-streaks, so that if a swimmer swims faster than expected on one day, they do so the next day too because of increased morale.

One way to formalize the two sources of randomness is through a hierarchical model. First, for $i = 1, \dots, 100$ (the 100 swimmers), we draw:

$$F_i \stackrel{\text{iid}}{\sim} \mathbb{P}, \quad (39)$$

where F_i is itself a probability measure on say \mathbb{R}^{41} and \mathbb{P} is a probability measure on the space of probability measures. Here we assume that all the swimmers come from the same population that is described by \mathbb{P} . At the second level of sampling, we draw for the i -th swimmer

$$(S_{it})_{1 \leq t \leq 41} \mid F_i \sim F_i, \quad (40)$$

and the statistician gets to observe $S_{it}, 1 \leq t \leq 40$ and seeks to predict the $S_{i,41}$.

Let us make this concrete by a simple random effects model. Say the only thing that differs across swimmers is the their mean speed μ_i . Then the sampling could be described by the two-level model:

$$\mu_i \sim \mathcal{N}(\nu, \tau^2), \quad S_{i\cdot} \mid \mu_i \sim \mathcal{N}(\mu_i, \Sigma),$$

for some ν, τ^2, Σ . This is not necessarily a good model, but it illustrates the more general two-level model described above; here the random distributions F_i are equal to $F_i = \mathcal{N}(\mu_i, \Sigma)$.

The difference between the two statisticians now can be described as follows: Statistician A cares about the uncertainty induced from the sampling step in (39), while statistician B about the randomness in (40).

Let us first discuss the two approaches assuming the model used to predict the 41st day is fixed and not trained based on the data at hand.

If we care about each swimmer individually, and want to assess uncertainty of the prediction individually for each swimmer, then we should follow the recommendation of Statistician B.¹²

We would prefer an approach that samples swimmers (as suggested by Statistician A) if we care about the population of swimmers. For example, we could answer the

¹²As an aside, it is perhaps worth noting a peculiarity in a time-series bootstrap when viewed in light of (40). We seek to represent uncertainty in F_i based on a *single* sample from F_i ! This is only possible under strong assumptions, e.g., a parametric model assumption or the assumption that measurements taken multiple days apart are approximately independent.

following question: What is the average predicted speed across all possible swimmers in our population and what is our uncertainty? Then, we could resample swimmers with replacement (i.e., get $(S_{i_j.})_{1 \leq j \leq 100}$ with $i_j \in \{1, \dots, 100\}$), get the prediction for each, average these predictions to get an average prediction. We could repeat the sampling step multiple times and finally report a bootstrap percentile interval.

The suggestion of statistician A would also make more sense if the model is trained on the swimmers themselves. For simplicity, let us consider a leave-one-out prediction scheme in which we are interested in predicting the speed of swimmer 1 on day 41 and train a model based on the other 99 swimmers. Then, keeping the time-series of swimmer 1 fixed, there is still randomness in the prediction because of the random selection of the other 99 swimmers. We could assess this uncertainty by resampling the 99 swimmers with replacement, building a new model each time and predicting the final speed of swimmer 1. An alternative here would be to combine the approaches of statisticians A and B: we could resample the other 99 swimmers with replacement to build a new predictive model, then use the time-series bootstrap to resample the data of swimmer 1, and finally predict the final speed of swimmer 1. Such a scheme would account for both sources of uncertainty.

Problem 6: Population LASSO

Key idea/tools:

- Approximating the population objectives through empirical averages.
- The LASSO fit depends only on $\mathbf{X}^\top \mathbf{Y}$ and $\mathbf{X}^\top \mathbf{X}$.
- Decomposing a covariance matrix through the Cholesky factorization or its eigenvalue decomposition.

In this problem we are interested in solving the population LASSO problem,

$$\operatorname{argmin}_{\beta} \frac{1}{2} \mathbb{E} \left[(Y - \mathbf{X}^\top \beta)^2 \right] + \lambda \|\beta\|_1, \quad (41)$$

where (X, Y) are centered and have a known covariance matrix Σ . Instead, we have at our disposal a LASSO solver for the empirical objective:

$$\operatorname{argmin}_{\beta} \frac{1}{2} \|\mathbf{Y} - \mathbf{X}^\top \beta\|_2^2 + \lambda \|\beta\|_1, \quad (42)$$

where \mathbf{X} is a $N \times p$ design matrix with rows x_i^\top and \mathbf{Y} a $N \times 1$ response vector with entries y_i .

(a) Upon expanding the square of the population mean-squared error, we find that

$$\mathbb{E} \left[(Y - \mathbf{X}^\top \beta)^2 \right] = \mathbb{E} \left[Y^2 + \beta^\top \mathbf{X} \mathbf{X}^\top \beta - 2Y \mathbf{X}^\top \beta \right] = \sigma_{YY} + \beta^\top \Sigma_{XX} \beta - 2\sigma_{XY}^\top \beta \quad (43)$$

In particular, the solution of the optimization problem only depends on the covariance Σ of (X, Y) . So for $i = 1, \dots, N$ we could draw samples (x_i, y_i) from a multivariate normal distribution with mean 0 and the given covariance Σ . Then we could solve:

$$\min_{\beta} \frac{1}{2N} \sum_{i=1}^N (y_i - x_i^\top \beta)^2 + \lambda \|\beta\|_1 \quad (44)$$

For large enough N , by the law of large numbers, this would provide a good approximation to the population LASSO objective, since:

$$\frac{1}{N} \sum_{i=1}^N (y_i - x_i^\top \beta)^2 \approx \mathbb{E} [y_i - x_i^\top \beta]^2,$$

where in light of (43) the approximation can be made uniform in β (say for β in a compact set; and we can check that for $\lambda > 0$ the optimal β for the population problem indeed lies in a compact set).

To use our friend's software we would use the pairs (x_i, y_i) as input and set the penalty parameter to $N \cdot \lambda$.

Let us provide a few more details on drawing samples from the specified multivariate normal distribution. Let $\Sigma = LL^\top$ be the Cholesky decomposition of Σ ¹³ Then let $u_i^j \stackrel{\text{iid}}{\sim} \mathcal{N}(0, 1)$, $j = 1, \dots, p+1$, $u_i = (u_i^1, \dots, u_i^{p+1})$ and set:

$$(x_i, y_i) = z_i = Lu_i$$

Then indeed $(x_i, y_i) \sim \mathcal{N}(0, \Sigma)$.

- (b) This question is phrased in a somewhat confusing way. My interpretation is as follows: let us assume we can solve (44) exactly, i.e., let us ignore floating point arithmetic errors. Then, can we use (44) to exactly solve (41)? It turns out that we can achieve this with what amounts to the computational shortcut for kernel learning [Hastie and Tibshirani, 2004] when $p > n$.

Let $\Sigma = LL^\top$. This could be for example the Cholesky decomposition of Σ (as explained for sampling in part a; also see footnote therein). Then let \mathbf{Y} be the last row of L and \mathbf{X}^\top the $p \times p+1$ matrix consisting of the first p rows of L . We can think of this procedure as providing as with pseudoobservations (x_i, y_i) .

Note that by expanding as we did in (43) for the population objective, we get

$$\|\mathbf{Y} - \mathbf{X}^\top \beta\|_2^2 = \|\mathbf{Y}\|_2^2 - 2\mathbf{Y}^\top \mathbf{X} \beta + \beta \mathbf{X}^\top \mathbf{X} \beta \quad (45)$$

However, by construction, we have that $\mathbf{X}^\top \mathbf{X} = \Sigma_{XX}$, $\|\mathbf{Y}\|_2^2 = \sigma_Y$ and $\mathbf{X}^\top \mathbf{Y} = \sigma_{XY}$, so that these problems are the same.

¹³Instead of the Cholesky decomposition, we could have used the Eigendecomposition $\Sigma = VD^2V^\top$ and then proceeded as above with $L = VD$.

9 Applied 2020: Solution¹⁴

Problem 1: Maximum Likelihood for Truncated Data

- a) Both `t.test(Y)` and `lm(Y ~ 1)` compute the mean and standard deviation Y . This is not possible without observing the full dataset. By default, both of these functions will simply ignore the missing values. This will cause us to overestimate the mean of Y and produce a confidence interval that is biased upwards.
- b) The methods in part a) give estimates that have a positive bias. To derive a likelihood based method let Z_1, \dots, Z_n denote the latent values of the light emissions. We are told that $Z_i \sim N(\mu, \sigma^2)$ for some unknown values μ and σ^2 , and we observe

$$Y_i = \begin{cases} Z_i, & \text{if } Z_i > C \\ \text{NA}, & \text{otherwise} \end{cases}.$$

Let Φ and ϕ denote the cdf and pdf of $N(0, 1)$, respectively. Let $\delta_i := \mathbb{1}_{Y_i \neq \text{NA}}$. Then, the likelihood of the observed data is

$$p(Y; \mu, \sigma) = \prod_{i=1}^n \left(\frac{1}{\sigma} \phi \left(\frac{Y_i - \mu}{\sigma} \right) \right)^{\delta_i} \Phi \left(\frac{C - \mu}{\sigma} \right)^{1 - \delta_i}$$

and, up to a constant, the log-likelihood is

$$\ell(\mu, \sigma) = \sum_{i=1}^n -\delta_i \left(\log(\sigma) - \frac{1}{2} \left(\frac{Y_i - \mu}{\sigma} \right)^2 \right) + (1 - \delta_i) \log \Phi \left(\frac{C - \mu}{\sigma} \right).$$

This is not a concave function of (μ, σ) , but it is concave in the transformed parameter $\lambda = \frac{1}{\sigma}$ and $\nu = \frac{\mu}{\sigma}$. In these parameters, we have

$$\ell(\lambda, \nu) = \sum_{i=1}^n \delta_i \left(\log(\lambda) - \frac{1}{2} (\lambda Y_i - \nu)^2 \right) + (1 - \delta_i) \log \Phi(\lambda C - \nu).$$

This function is concave because is non-negative combination of concave functions. We can thus use Newton's method to find $\hat{\lambda}, \hat{\nu}$ which we can transform to get $\hat{\sigma} = \frac{1}{\hat{\lambda}}$ and $\hat{\mu} = \frac{\hat{\nu}}{\hat{\lambda}}$.

If you do not recognize the above transformation, a good alternative is to use EM. In the E-Step we would need to compute

$$\mathbb{E}_{\hat{\mu}^t, \hat{\sigma}^t}[\log(p(Z; \mu, \sigma)) | Y].$$

By expanding $\log(p(Z; \mu, \sigma))$ we will find that it is sufficient to compute

$$\mathbb{E}_{\hat{\mu}^t, \hat{\sigma}^t}[Z_i | Y_i] \quad \text{and} \quad \mathbb{E}_{\hat{\mu}^t, \hat{\sigma}^t}[Z_i^2 | Y_i].$$

¹⁴Isaac Gibbs, Dan Kluger and M.H.

If $Y_i \neq NA$ then $\mathbb{E}[Z_i|Y_i] = Y_i$ and $\mathbb{E}[Z_i^2|Y_i] = Y_i^2$. Otherwise, computing these expectations reduces to computing the mean and variance of a truncated Gaussian, which is a straightforward computation that we could carry out. Finally, in the M-step we have to optimize $\mathbb{E}_{\hat{\mu}^t, \hat{\sigma}^t}[\log(p(Z; \mu, \sigma))|Y]$ over μ and σ , which is a straightforward two dimensional calculus problem which yields:

$$\hat{\mu}^{t+1} = \frac{1}{n} \sum_{i=1}^n \mathbb{E}_{\hat{\mu}^t, \hat{\sigma}^t}[Z_i|Y_i] \quad \text{and} \quad (\hat{\sigma}^{t+1})^2 = \frac{1}{n} \sum_{i=1}^n \left(\mathbb{E}_{\hat{\mu}^t, \hat{\sigma}^t}[Z_i^2|Y_i] - 2\hat{\mu}^{t+1} \mathbb{E}_{\hat{\mu}^t, \hat{\sigma}^t}[Z_i|Y_i] + (\hat{\mu}^{t+1})^2 \right)$$

- c) Intuitively we should have that the observed data Y_1, \dots, Y_n come from a truncated normal distribution. Thus, we could maximize the likelihood

$$p(Y; \mu, \sigma) = \prod_{i=1}^n \frac{1}{\sigma} \phi\left(\frac{Y_i - \mu}{\sigma}\right) \left(1 - \Phi\left(\frac{C - \mu}{\sigma}\right)\right)^{-1}.$$

The log of this likelihood is no longer concave in $\lambda = \frac{1}{\sigma}$ and $\nu = \frac{\mu}{\sigma}$, and so we have to use an inexact method such a gradient descent with multiple initialization. Another option would be to do a grid search over (μ, σ) .

Without doing any rigorous calculations (assuming both devices take measurements for the same amount of time) we suspect that the device in b) should give a better estimate of μ and σ than the device in c). This is due to the fact that in b) we observe strictly more data than in c).

Aside: The question explicitly says that we do not need to do a rigorous calculation. However, seeing a rigorous calculation could still be informative. This calculation shows that the likelihood written out above is actually the conditional likelihood for the Y_i , conditional on the number of observations that we observe.

The calculation proceeds as follows. Let Z_1, \dots, Z_m denote the latent non-truncated measurements of light brightness. Then, we observed Z_{i_1}, \dots, Z_{i_n} where $1 \leq i_1 < i_2 < \dots < i_n \leq m$ are the indices at which $Z_{i_j} > C$. Note that n is itself random here. Now, for any $x > C$ and $1 \leq i \leq n$ we have that

$$\begin{aligned} \mathbb{P}(Y_i > x | n = k) &= \sum_{i_1, \dots, i_k} \mathbb{P}(Y_i > x | n = k, Z_{i_1}, \dots, Z_{i_k} > C, Z_j \leq C, \forall j \notin \{i_1, \dots, i_k\}) \\ &\quad \cdot \mathbb{P}(Z_{i_1}, \dots, Z_{i_k} > C, Z_j \leq C, \forall j \notin \{i_1, \dots, i_k\} | n = k). \end{aligned}$$

Given fixed values for the indices i_1, \dots, i_k let j^* be the index such that $Y_i = Z_{i_{j^*}}$.

Then,

$$\begin{aligned}
& \mathbb{P}(Y_i > x | n = k, Z_{i_1}, \dots, Z_{i_k} > C, Z_j \leq C, \forall j \notin \{i_1, \dots, i_k\}) \\
&= \frac{\mathbb{P}(Z_{j^*} > x, Z_{i_1}, \dots, Z_{i_k} > C, Z_j \leq C, \forall j \notin \{i_1, \dots, i_k\})}{\mathbb{P}(Z_{i_1}, \dots, Z_{i_k} > C, Z_j \leq C, \forall j \notin \{i_1, \dots, i_k\})} \\
&= \frac{(1 - \Phi(\frac{x-\mu}{\sigma})) \Phi(\frac{C-\mu}{\sigma})^{m-k} (1 - \Phi(\frac{C-\mu}{\sigma}))^{k-1}}{\Phi(\frac{C-\mu}{\sigma})^{m-k} (1 - \Phi(\frac{C-\mu}{\sigma}))^k} \\
&= \frac{1 - \Phi(\frac{x-\mu}{\sigma})}{1 - \Phi(\frac{C-\mu}{\sigma})}.
\end{aligned}$$

Differentiating this last expression gives the claimed formula for the likelihood.

- d) We are told to replace μ in the above likelihoods with $\mu_i = X_i^\top \beta$. Since we are given a function for the gradient of the log-likelihoods we can use gradient ascent to maximize the objective. Assuming that σ^2 is fixed and known the updates would look like

$$\beta_{t+1} = \beta_t + \eta \text{grad_log}(Y, X\beta_t, \text{sigma_sq}, C)^T X, \quad (46)$$

where $\eta > 0$ is a fixed step-size parameter. If σ^2 is unknown then we would still use the update (46), but in this update we would only use the first n coordinates of $\text{grad_log}(Y, X\beta_t, \text{sigma_sq}, C)$, we would replace sigma_sq with σ_t^2 , and we would additionally have the update

$$\sigma_{t+1} = \sigma_t + \eta \left[\text{grad_log}(Y, X\beta_t, \sigma_t^2, C) \right]_{n+1}.$$

Finally, we could use the function $\text{logl}(Y, \mu, \text{sigma_sq}, C)$ to judge the results of gradient ascent with multiple different random initializations.

Problem 2: Low-Rank Matrix Factorization

- a) The goal is to find matrices $A \in \mathbb{R}^{T \times 13}$ and $B \in \mathbb{R}^{990 \times 13}$ that minimize $\|X - AB^T\|_2^2$. We know that the skinny SVD optimizes this objective. Namely, let

$$X = \sum_{i=1}^{990} \sigma_i u_i v_i^T$$

denote the SVD of X where $|\sigma_1| \geq \dots \geq |\sigma_{990}| \geq 0$. Then, we can take

$$\hat{A} = U_{1:13} \text{diag}(\sigma_1, \dots, \sigma_{13}) \quad \text{and} \quad \hat{B}^T = (V_{1:13})^T$$

where $U_{1:13}$ and $V_{1:13}$ denotes the matrices with columns u_1, \dots, u_{13} and v_1, \dots, v_{13} , respectively.

- b) The individual matrices A and B will not be unique. Namely, given an optimal solution (A, B) we have that for any orthogonal matrix O , $AB^T = AOO^TB^T$ and thus (AO, BO) is also a solution. On the other hand, AB^T will be unique iff $\sigma_{13} > \sigma_{14}$ since in this case the skinny SVD is unique.
- c) Let A_1, \dots, A_T denote the rows of A and B_1, \dots, B_{990} denote the rows of B . Then,

$$\|X - AB^T\|_2^2 = \sum_{i=1}^T \sum_{j=1}^{990} (X_{ij} - B_j^T A_i)^2.$$

For each fixed value of $i \in \{1, \dots, T\}$ we recognize $\sum_{j=1}^{990} (X_{ij} - B_j^T A_i)^2$ as a linear regression problem with feature-response pairs $\{(B_j, X_{ij})\}_{1 \leq j \leq 990}$. Using known results from linear regression this expression will be minimized by taking

$$A_i = (B^T B)^{-1} B^T x_i.$$

By the separability of the objective function in i it follows that for a fixed B , A is optimized when setting $A^T = (B^T B)^{-1} B^T X^T$. Note that this assumes that B is full-rank. If it is not full rank then you can replace the inverse by a pseudo-inverse.

- d) Using the same reasoning as in part c) we have that the minimizing value for B will be given by

$$B^T = (A^T A)^{-1} A^T X.$$

Note that this assumes that A is full-rank. If it is not full rank then you can replace the inverse by a pseudo-inverse.

- e) Letting $x_{T+1, \mathcal{O}} \in \mathbb{R}^{990-45}$, denote the vector of observations at time $T+1$ with the entries dropped and let $B_{\mathcal{O}} \in \mathbb{R}^{(990-45) \times 13}$ denote the matrix B with the rows which correspond to missing entries of x_{T+1} removed. We could estimate a_{T+1} by setting

$$\hat{a}_{T+1} = \arg \min_a \|x_{T+1, \mathcal{O}} - B_{\mathcal{O}} a\|_2^2,$$

As in parts c) and d) solving this optimization problem is equivalent to solving a linear regression problem and obtain that $\hat{a}_{T+1} = (B_{\mathcal{O}}^T B_{\mathcal{O}})^{-1} B_{\mathcal{O}}^T x_{T+1, \mathcal{O}}$. Then, we could estimate the missing values in the full vector by looking at the corresponding entries of $\hat{x}_{T+1} = B \hat{a}_{T+1}$.

Problem 3: Poisson GLMs

- a) You can take X to be the matrix with n_i copies of row x_i and $Y \in \mathbb{R}^N$ to be the vector with entries y_{ij} and then use the R call `glm(Y ~ X - 1, family=poisson(link="log"))`.

b) The likelihood for the data is

$$p(Y; X, \beta) = \prod_{ij} \frac{\exp(y_{ij}x_i^T\beta - \exp(x_i^T\beta))}{y_{ij}!} \propto \prod_i \exp(\sum_j y_{ij}x_i^T\beta - n_i \exp(x_i^T\beta)).$$

Additionally, note that $T_i := \sum_j y_{ij} \sim \text{Poisson}(n_i \exp(x_i^T\beta))$. So, the likelihood for T_1, \dots, T_I is

$$p(T_1, \dots, y_I; x_1, \dots, x_I, \beta) \propto \prod_i \exp(T_i x_i^T \beta - n_i \exp(x_i^T \beta)).$$

In particular, we find that the likelihood for the data $\{T_i\}$ is proportional to the likelihood for $\{y_{ij}\}$. Thus, maximum likelihood based inference will be the same for these two datasets. Under, the current model we have that the mean of T_i is μ_i . with

$$\log(\mu_i) = \log(n_i) + x_i^T \beta$$

We recognize this as a GLM with offsets. Let $T = (T_1, \dots, T_I) \in \mathbb{R}^I$, $\tilde{X} \in \mathbb{R}^{I \times p}$ be the matrix with rows x_i , and $n = (n_1, \dots, n_I)$. Then, we find that the model from part a) can be fit using the GLM call `glm(T ~ $\tilde{X}-1$, family=poisson(link="log"), offset=log(n))`.

Note that your justification above need not explicitly write out the likelihood and could use a sufficient statistics argument instead.

c) Extending our previous model it may now be reasonable to posit that the events for galaxy i come from a homogeneous Poisson process with mean parameter $\exp(x_i^T\beta)$. i.e. we have that y_{ij} are independent Poisson random variables with

$$\log(\mu_{ij}) = \log(\ell_{ij}) + x_i^T \beta.$$

We can fit this model using the call `glm(Y ~ X-1, family=poission(link="log"), offset = log(l))` where ℓ is the vector of lengths and Y and X are as in part a).

Problem 4: Estimating Starfish Diversity

a) S_i will be larger for less diverse antibody pools. One way to see this is to observe that

$$S_i = \sum_{j=1}^J \hat{p}_{ij}^2 \leq \sum_{j=1}^J \hat{p}_{ij} = 1,$$

with equality for vectors \hat{p}_i . that satisfy $\hat{p}_{ij} = 1$ for some j and $\hat{p}_{ij'} = 0$ for all other $j' \neq j$. i.e. S_i is maximized by the least diverse antibody pools. Moreover, by Jensen's inequality we have that

$$S_i = J \sum_{j=1}^J \frac{\hat{p}_{ij}^2}{J} \geq J \left(\sum_{j=1}^J \frac{\hat{p}_{ij}}{J} \right)^2 = \frac{1}{J}$$

with equality being obtained by the uniform distribution. i.e. S_i is minimized by the most diverse populations.

- b) Perhaps the largest issue with S_H is that its value can be dominated by a single starfish that has very large counts for all antibodies. This is a major concern because we are told that "the absolute numbers n_{ij} depend a lot on how the sample was taken, and so the ratios are considered more useful." For a concrete example, suppose 9/10 of the starfish collected have very diverse antibodies and a relatively small total number of antibodies measured and the final remaining starfish does not have very diverse antibodies, but has a large total number of antibodies. Then, the value of S_H will be dominated by the one non-diverse starfish and we will estimate that the population does not have a very diverse set of antibodies even though 9/10s of the starfish have a large diversity. Cases like this where some samples have larger total counts than others are common in many types of biological data (e.g. RNA-seq).

A better method would be to weight all of the starfish equally regardless of the magnitude of their total counts. To do this we could define

$$S_{H,i} = \sum_{j=1}^J \left(\frac{n_{ij}}{\sum_{j'=1}^J n_{ij'}} \right)^2$$

to be the diversity measure for the i_{th} high-salinity starfish and then define the new estimator

$$\tilde{S}_H = \frac{1}{10} \sum_{i=1}^{10} S_{H,i}.$$

We could do the same thing for the low salinity starfish and compute the estimate $\tilde{S}_H - \tilde{S}_L$.

- c) Their bootstrap procedure does not appear to reflect the data generating mechanism and account for the clustered nature of their data. In particular, their bootstrap procedure treats each observed antibody in the high-salinity water as a sample, and samples the $\sum_j n_{H,j}$ antibodies with replacement. It does not account for the fact that the antibodies were measured by taking sampling antibodies in 10 different starfish (each of which could have different proportions of each antibody).

A better approach would be to use clustered bootstrap which resamples entire starfish with replacement but does not resample data within starfish (See Strategy 1 of Section 3.8 in Davison and Hinkley [1997]). You can think of this as a block bootstrap where each starfish is a block. More specifically, for each $b = 1, \dots, B$ we would obtain bootstrap datasets of starfish-level diversity scores $\{S_{H,i}^b\}_{1 \leq i \leq 10}$ and $\{S_{L,i}^b\}_{1 \leq i \leq 10}$ where $S_{H,i}^b \sim \text{Unif}(S_{H,1}, \dots, S_{H,10})$ and

$S_{L,i}^b \sim \text{Unif}(S_{L,1}, \dots, S_{L,10})$. We would then use these datasets to get new estimates of the difference in means given by

$$\tilde{S}_H^b - \tilde{S}_L^b = \frac{1}{10} \sum_{i=1}^{10} S_{H,i}^b - \frac{1}{10} \sum_{i=1}^{10} S_{L,i}^b$$

and then form a confidence interval by looking at the empirical quantiles of $\{\tilde{S}_H^b - \tilde{S}_L^b\}_{1 \leq b \leq B}$ (see Section ??).

An alternative to the previous procedure is to use a clustered bootstrap, where you sample starfish with replacement, and subsequently resample the antibodies within each starfish with replacement (See Strategy 2 of Section 3.8 in Davison and Hinkley [1997]). In particular, for $b = 1, \dots, B$, to construct the bootstrap dataset for the high-salinity starfish, sample 10 high-salinity with replacement $i_1, \dots, i_{10} \stackrel{IID}{\sim} \text{Unif}\{1, \dots, 10\}$ then for $k = 1, \dots, 10$, compute $S_{H,k}^b$, by resampling the antibodies from starfish i_k with replacement (this can be done by sampling a multinomial on $\{1, \dots, J\}$ with probabilities $\hat{p}_{i_k,j}$ with $\sum_j n_{i_k,j}$ trials for the multinomial) and then using the resampled antibody counts in starfish i_k to compute the diversity score $S_{H,k}^b$. You would do the same procedure to generate bootstrap samples of the diversity scores $S_{L,k}^b$ in the Low salinity waters. Finally you would let the b th bootstrap statistic be

$$\tilde{S}_H^b - \tilde{S}_L^b = \frac{1}{10} \sum_{k=1}^{10} S_{H,k}^b - \frac{1}{10} \sum_{k=1}^{10} S_{L,k}^b,$$

and use the empirical quantiles of $\{\tilde{S}_H^b - \tilde{S}_L^b\}_{1 \leq b \leq B}$ to compute a bootstrap confidence interval.

While this is out of scope for quals, Section 3.8 in Davison and Hinkley [1997] recommends Strategy 1 over Strategy 2, but either would be an acceptable answer for quals.

Problem 5: EM for a Mixture of Regressions

a) We have that

$$\log(p_\theta(Y)) = \sum_{i=1}^n \log \left(\sum_{j=1}^k \pi_j \frac{1}{\sqrt{2\pi\sigma^2}} \exp \left(-\frac{1}{2\sigma^2} (Y_i - a_j - b_j X_i)^2 \right) \right)$$

and

$$\begin{aligned}
\log(p_\theta(Y, Z)) &= \sum_{i=1}^n \log \left(\pi_{Z_i} \frac{1}{\sqrt{2\pi\sigma^2}} \exp \left(-\frac{1}{2\sigma^2} (Y_i - a_{Z_i} - b_{Z_i} X_i)^2 \right) \right) \\
&= \sum_{i=1}^n \log(\pi_{Z_i}) - \frac{1}{2\sigma^2} (Y_i - a_{Z_i} - b_{Z_i} X_i)^2 - \frac{1}{2} \log(2\pi\sigma^2) \\
&= \sum_{i=1}^n \left(\sum_{j=1}^k \left(\log(\pi_j) Z_{ij} - \frac{1}{2\sigma^2} (Y_i - a_j - b_j X_i)^2 Z_{ij} \right) - \frac{1}{2} \log(2\pi\sigma^2) \right).
\end{aligned}$$

There are k independent parameters for the a_j 's, k independent parameters for the b_j 's, $k-1$ independent parameters for the π_j , and one parameter for σ for a total of $3k$ independent parameters.

- b) The notation here isn't very good. In order to run EM we will need to compute $\mathbb{E}_{\tilde{\theta}}[\log(p_\theta(Y, Z))]$ not $\mathbb{E}_\theta[\log(p_\theta(Y, Z))]$ where $\tilde{\theta}$ and θ are two potential values for the fitted parameters. Thus, I will re-define $\tau_{ij} = \mathbb{P}_{\tilde{\theta}}(Z_i = j|Y)$. This gives the expression

$$\mathbb{E}_{\tilde{\theta}}[\log(p_\theta(Y, Z))] = \sum_{i=1}^n \left(\sum_{j=1}^k \left(\log(\pi_j) \tau_{ij} - \frac{1}{2\sigma^2} (Y_i - a_j - b_j X_i)^2 \tau_{ij} \right) - \frac{1}{2} \log(2\pi\sigma^2) \right). \quad (47)$$

- c) The description of EM given in the problem statement is quite bad. The expression $\mathbb{E}_{\theta, \tau}[\cdot]$ doesn't really make any sense and should be $\mathbb{E}_{\tilde{\theta}, \tau}[\cdot]$. Regardless we will solve the problem ignoring the notational issues using the correct version of the EM algorithm. The goal is to optimize (47) over (a, b, π, σ) (strangely the question does not ask us to optimize π , but π is part of θ so we certainly need to optimize over it). Optimizing over π is a straightforward Lagrange multipliers calculation from which you should find that

$$\hat{\pi}_j = \frac{\sum_{i=1}^n \tau_{ij}}{\sum_{k=1}^K \sum_{i=1}^n \tau_{ik}} = \frac{\sum_{i=1}^n \tau_{ij}}{n},$$

where one easily checks that from the definitions we must have that $\sum_{j=1}^K \tau_{ij} = 1$ for all i . Optimizing a_j and b_j splits into K standard weighted least squares problems. Let $T_j = \text{diag}(\tau_{1j}, \dots, \tau_{nj})$ and let $X = [\mathbf{1} \ (X_i)_{i=1}^n]$. Then, we have that

$$\begin{aligned}
\begin{pmatrix} \hat{a}_j \\ \hat{b}_j \end{pmatrix} &= (X^T T_j X)^{-1} X^T T_j Y \\
&= \frac{1}{(\sum_{i=1}^n \tau_{ij} X_i^2)(\sum_{i=1}^n \tau_{ij}) - (\sum_{i=1}^n \tau_{ij} X_i)^2} \\
&\quad \cdot \begin{pmatrix} (\sum_{i=1}^n \tau_{ij} X_i^2)(\sum_{i=1}^n \tau_{ij} Y_i) - (\sum_{i=1}^n \tau_{ij} X_i)(\sum_{i=1}^n \tau_{ij} X_i Y_i) \\ (\sum_{i=1}^n \tau_{ij})(\sum_{i=1}^n \tau_{ij} X_i Y_i) - (\sum_{i=1}^n \tau_{ij} X_i)(\sum_{i=1}^n \tau_{ij} Y_i) \end{pmatrix}
\end{aligned}$$

Finally, by differentiating in terms of σ^2 and re-arranging one can easily compute that

$$\hat{\sigma}^2 = \frac{1}{n} \sum_{i=1}^n \sum_{j=1}^k (Y_i - \hat{a}_j - \hat{b}_j X_i)^2 \tau_{ij}.$$

- d) The correct way to compute BIC is to use the **observed** data log-likelihood. This gives the value

$$\text{BIC} = 3k \log(n) + 2 \cdot 116.36 = 9 \cdot \log(39) + 2 \cdot 116.36 \approx 265.7.$$

- e) In our setting there are $n = 39$ samples and $k = 3K$ independent parameters, so we can use the definition of AIC and BIC to compute the AIC and BIC values for $K \in \{1, \dots, 5\}$ in the table presented below. AIC is minimized at $K = 5$ so AIC selects $K = 5$ (or it could select $K > 5$ depending on un-presented log likelihood values). On the other hand the BIC is minimized at and selects $K = 4$. There are many different alternative ways to select the number of clusters. One option is to use information in the forestry literature to set a prior on K and also a prior on $\theta \mid K$ and then compute the mode of the posterior distribution of K given the data.

Method	Formula	$K = 1$	$K = 2$	$K = 3$	$K = 4$	$K = 5$
AIC	$6K - 2 \log p_{\hat{\theta}}(Y)$	294.08	270.38	250.72	241.56	239.16
BIC	$3K \log(39) - 2 \log p_{\hat{\theta}}(Y)$	299.0707	280.3614	265.6921	261.5227	264.1134

Problem 6: Estimating the Mutation Rate

There are many possible models you could consider. Two obvious choices are a Poisson model and a linear model. In what follows I will answer parts a, b, and c separately for each of these two choices.

Solution using a Poisson model

One potential model is that the $d_{i,r}$ values are independent with

$$d_{i,r} | t_i - t_r \sim \text{Poisson}((t_i - t_r)\mu).$$

One nice aspect of this model is that the variance of $d_{i,r}$ scales linearly in the mean, which seems consistent with what we observe in the plot of the data. A straightforward calculation shows that this gives the maximum likelihood estimator

$$\hat{\mu} = \frac{\sum_{i=1}^n d_{i,r}}{\sum_{i=1}^n t_i - t_r}.$$

This estimator is unbiased for μ and since it is a sum of independent random variables it should also be consistent under only mild assumptions on the times $t_i - t_r$. To do inference for μ we could form the standard confidence interval for the MLE. This would be valid as long as the Poisson model is accurate. If we want to avoid this parametric assumption an alternative is to use the fact that $\hat{\mu}$ is unbiased and an average of independent random variables and thus directly derive a CLT for $\hat{\mu}$. The main thing we will need here is an estimate of the variance of $\hat{\mu}$. We have that

$$\text{Var}(\hat{\mu}) = \frac{1}{(\sum_{i=1}^n t_i - t_r)^2} \sum_{i=1}^n \text{Var}(d_{i,r}).$$

We know that $\text{Var}(d_{i,r}^2) = \mathbb{E}[(d_{i,r} - (t_i - t_r)\mu)^2]$. Since $\hat{\mu}$ is consistent for μ a reasonable way to estimate this quantity is using the estimator

$$\hat{\sigma}^2 = \frac{1}{(\sum_{i=1}^n t_i - t_r)^2} \sum_{i=1}^n (d_{i,r} - (t_i - t_r)\hat{\mu})^2.$$

Finally we can use the normal approximation $\hat{\mu} \sim N(\mu, \hat{\sigma}^2)$ to get a confidence interval for μ .

One flaw in our model is that it assumes a Poisson distribution for the mutations. However, above we derived a non-parametric method for computing a confidence interval for μ and thus this modelling assumption is not critical. A potentially larger issue is the presence of outliers in the dataset. In particular, we see that at some of the larger time points there are a few outlying points with very large numbers of mutations (note that a $\text{Pois}(10)$ Random variable is 40 or larger with probability less than 10^{-12}). These points could have a large influence on the estimator $\hat{\mu}$ and thus heavily impact our estimate of the mutation rate. As a first step we should investigate how the data was collected and try to determine if there were any possible errors in the data collection process that could lead to these outlying points. Alternatively, maybe there is a biological mechanism that would tell us that some rare cases will deviate from the Poisson model presented. In both of these cases we might want to remove these outlying points from the dataset before fitting the model. If we think these outlying points are true data points that cannot be ignored then we could still use the above estimator $\hat{\mu}$. However, we should be conscious of the fact that these points may have an outsized influence on the estimator that will invalidate our normal approximation. To try to judge the size of the influence of these points we could compute $\hat{\mu}$ both with and without removing the outliers and report both values.

Another issue with our model assumption is the assumption of independent observations. In particular, it could be that many observations had a recent ancestor that is also in the data. For example, an observation in January 2020 may be correlated with its direct descendants in the data (if a virus observed in January 2020 has a relatively high hamming distance from the original strain, it is likely the descendants of that virus also have a relatively high hamming distance from the original strain). It

is hard to tell from the information given whether there will be a lot of dependency between the samples without domain knowledge or additional information beyond $d_{r,i}$, t_i and t_r . That being said, even if the samples are correlated, the proposed estimator for μ should still be unbiased and consistent, but the confidence intervals would likely be too small.

Solution using a linear model

An alternative to fit a linear model

$$d_{i,r} = (t_i - t_r)\mu + \epsilon_i.$$

Intuitively, it is reasonable to expect that the variance of ϵ_i will be larger for larger times as there is more time for mutation events to occur and thus more time to accumulate deviations from the mean. From the plot of the data it looks like a reasonable model would be that $\text{Var}(\epsilon_i)$ increases linearly with time. Thus, we could model that $\text{Var}(\epsilon_i) = (t_i - t_r)\sigma^2$. Using weighted least squares this would give the estimator

$$\hat{\mu} = \frac{\sum_{i=1}^n d_{i,r}}{\sum_{i=1}^n (t_i - t_r)},$$

which is the same as the estimator from the Poisson model. All the considerations from the previous section still apply to this estimator. Here it is even more clear that we should not make standard modelling assumptions like $\epsilon_i \sim N(0, (t_i - t_r)\sigma^2)$ since we know that the $d_{i,r}$ are discrete. The normal approximation given in the previous section could still be a good way to form a confidence interval for μ .

10 Applied 2021: Solution¹⁵

Key Ideas/ Main Tools: Unmeasured confounders, GLMs, offsets

Problem 1: Modeling association between vaccination and death rates for Covid-19

- (a) It is essential to ask the researchers whether they know the population in each county because counties with higher populations will tend to have a higher death count irrespective of the vaccination rates. Also, if rural counties with low populations tend to have low vaccination rates, not accounting for county population can make vaccination seem less effective than it is.

Other questions worth asking the researchers are if there are any confounding variables that are likely to affect both vaccination rates and death rates, and if any of those confounding variables are measured. For example, the quality of the health care system in a county would affect both vaccination rates and deaths, so it would be worth asking if there are any measured variables that reflect the quality of the healthcare system in each county. Another example is the age demographics. The age demographics can influence both the death rate and the vaccination rate in a county, so it would be helpful to know if the researchers have any covariates that reflect age demographics (e.g. the percentage of the population above age 70 in each county). If the researchers mention many confounding variables that they have measurements for, then I would ask them to select the few most important ones based on their domain knowledge. I would mention that they should choose much fewer than 20 variables to control for because there are only 20 samples.

In addition to asking about county population levels and whether there are measured confounder variables not presented in the table, it would be worthwhile to check with the researcher that the deaths were counted after the vaccines were distributed (otherwise any analysis would be unable to say anything about the effect of vaccination rates on death rates).

- (b) Suppose the researchers are able to provide you with the population counts N_1, \dots, N_{20} of the 20 counties, and for each of the 20 they can give you vectors z_1, \dots, z_{20} of the few most important measured confounder variables for each county (e.g. age demographics and health care system quality metrics). Also suppose that their death counts in each county, indeed only include deaths from a time period after most of the vaccinations were given.

Letting d_i denote the number of deaths in county i , v_i denote the vaccination rate, $x_i \equiv (v_i, z_i)$, I would fit the following Poisson GLM with offsets $\alpha_i \equiv \log(N_i)$

¹⁵Dan Kluger and M.H.

$$d_i \stackrel{\text{ind}}{\sim} \text{Poisson}(\mu_i) \quad \log(\mu_i) = \alpha_i + x_i^\top \beta \quad \text{for } i = 1, \dots, 20.$$

After fitting the GLM (which can easily be done in R) using the `glm` function, I would look at the confidence interval for the first estimated coefficient $\hat{\beta}_1$. If the confidence interval only contains negative values, then we can conclude that the data suggests higher vaccination rates are associated with lower death rates when controlling for the confounders encoded in the z_i .

A binomial GLM could also be appropriate

$$d_i \stackrel{\text{ind}}{\sim} \text{Binom}(N_i, p_i) \quad \text{Logit}(p_i) = \beta_0 + x_i^\top \beta \quad \text{for } i = 1, \dots, 20.$$

Again, the confidence interval around $\hat{\beta}_1$ will let us do inference on the affect of vaccination controlling for z_i .

Problem 2: Finding an essential subset

Key Ideas/ Main Tools: Group Lasso

Defining the optimal essential subset as a solution to an optimization problem

Observe that one way to obtain an essential subset is to find the matrix $B \in \mathbb{R}^{750 \times 750}$ that minimizes $\|R - RB\|_F^2$ subject to the constraint that only 25 of the rows of B are allowed to have nonzero entries. More formally, letting $B[i, \cdot]$ denote the i th row of the matrix B we could get an essential subset by solving the following optimization problem on B :

$$\boxed{\begin{array}{ll} \text{minimize} & \|R - RB\|_F^2 \quad \text{subject to} \quad \sum_{i=1}^{750} I\{B[i, \cdot] \neq \mathbf{0}\} \leq 25, B \in \mathbb{R}^{750 \times 750} \end{array}}.$$

Let \tilde{B} be the solution to the above optimization problem and define \mathcal{S} to be the set $\{i \in [750] : \tilde{B}[i, \cdot] \neq \mathbf{0}\}$. By definition, \mathcal{S} will give an essential subset, as each portfolio (represented by a column in R) will be reasonably well approximated by a linear combination of the portfolios of at most.

Solving the boxed optimization problem and setting \mathcal{S} to be the nonzero rows of the solution will recover an essential subset of size at most 25, but unfortunately the optimization problem is non-convex (it has an ℓ_0 type constraint).

A tractable approach using the Group Lasso

One way to induce a sparse number of nonzero rows of B is to use the Group Lasso. In particular, for each $\lambda > 0$, the Group Lasso can be used to solve the following convex optimization problem of finding:

$$\hat{B}_\lambda \in \underset{B \in \mathbb{R}^{750 \times 750}}{\operatorname{argmin}} \left(\frac{1}{2} \|R - RB\|_F^2 + \lambda \sum_{i=1}^{750} \|B[i, \cdot]\|_2 \right).$$

We can solve this group Lasso problem for many different λ values until we find a solution \hat{B}_λ which has exactly 25 rows which have nonzero. In particular, letting $\hat{\mathcal{S}}_\lambda = \{i \in [750] : \hat{B}_\lambda[i, \cdot] \neq \mathbf{0}\}$, we can do a bisection search on λ until we find a λ_* for which $|\hat{\mathcal{S}}_{\lambda_*}| = 25$. Then we can report $\hat{\mathcal{S}}_{\lambda_*}$ to our boss as an essential subset. Note that this may not be the optimal essential subset in the sense of minimizing $\|R - RB\|_F^2$ subject to 25 nonzero rows of B ; however, it will still be an essential subset according to your boss's definition (that any portfolio in not in $\hat{\mathcal{S}}_{\lambda_*}$ can be well approximated by a linear combination portfolios in $\hat{\mathcal{S}}_{\lambda_*}$).

Additional References: In some years, the Group Lasso is covered in the 305 coursework's lecture notes¹⁶, but it is not covered every year. See Obozinski et al. [2011] for a reference on the Group Lasso and some its theoretical guarantees in recovering a sparse set of rows.

Problem 3: Constructing Conformal Prediction Intervals

Key Ideas/ Main Tools: Prediction Intervals, Conformal Inference, Exchangeability

- (a) The defining property of the prediction interval is that $\mathbb{P}(Y_{n+1} \in [L, U]) \geq 1 - \alpha$, where \mathbb{P} is the joint distribution of the $n + 1$ data points $(X_i, Y_i)_{i=1}^{n+1}$.
- (b) This procedure is not reasonable because the more you overfit the data, the smaller the predictions intervals will be. Ideally our prediction will not be overconfident about overfit predictions. In an extreme case suppose that X_1, \dots, X_n are all distinct and that $\hat{\mu}$ is the best fit $n - 1$ degree polynomial to the first n datapoints. In this case, $\hat{\mu}(X_i) = y_i$ for all $i \in [n]$ implying that the residuals r_1, \dots, r_n are all equal to zero, further implying that the proposed prediction interval will have width zero. Clearly, if we overfit the data, the true prediction interval shouldn't have width zero for a new point X_{n+1} .
- (c) Let $\mathcal{S} = \{y \in \mathbb{R} : \pi(y) \leq (1 - \alpha)(n + 1)/n\}$ and let $L = \inf S$ and $U = \sup S$. To show that this gives a valid prediction interval first note that

$$\begin{aligned} \mathbb{P}(Y_{n+1} \in [L, U]) &\geq \mathbb{P}(Y_{n+1} \in \mathcal{S}) \\ &= \mathbb{P}(\pi(Y_{n+1}) \leq (1 - \alpha)(n + 1)/n) \\ &= \mathbb{P}\left(\frac{1}{n} \sum_{i=1}^n I\{R_{Y_{n+1}, i} \leq R_{Y_{n+1}, n+1}\} \leq (1 - \alpha)(n + 1)/n\right) \end{aligned}$$

¹⁶<https://web.stanford.edu/class/stats305c/notes/Regression/Sparse.html>

To simplify the above expression with an exchangeability argument, first define $\tilde{\mu}$ to be the curve fit to the $n + 1$ data points $(X_i, Y_i)_{i=1}^{n+1}$. Next, define for $i = 1, \dots, n + 1$, $V_i \equiv |Y_i - \tilde{\mu}(X_i)|$. Observe that since $(X_i, Y_i)_{i=1}^{n+1}$ are IID and $\tilde{\mu}$ is symmetric function of the collection of these $n + 1$ data points, $(V_i)_{i=1}^{n+1}$ is an exchangeable sequence of random variables. In addition, since $\tilde{\mu}(\cdot) = \hat{\mu}_{Y_{n+1}}(\cdot)$,

$$V_i \equiv |Y_i - \tilde{\mu}(X_i)| = |Y_i - \hat{\mu}_{Y_{n+1}}(X_i)| = R_{Y_{n+1}, i}.$$

Combining this with a previous result and using the exchangeability of $(V_i)_{i=1}^{n+1}$ (and assuming that almost surely $V_i \neq V_j$ for $i \neq j$),

$$\begin{aligned} \mathbb{P}(Y_{n+1} \in [L, U]) &\geq \mathbb{P}\left(\frac{1}{n} \sum_{i=1}^n I\{R_{Y_{n+1}, i} \leq R_{Y_{n+1}, n+1}\} \leq (1 - \alpha)(n + 1)/n\right) \\ &= \mathbb{P}\left(\frac{1}{n} \sum_{i=1}^n I\{V_i \leq V_{n+1}\} \leq (1 - \alpha)(n + 1)/n\right) \\ &= \mathbb{P}\left(\sum_{i=1}^n I\{V_i \leq V_{n+1}\} \leq (n + 1)(1 - \alpha)\right) \\ &\geq \mathbb{P}\left(\sum_{i=1}^n I\{V_i \leq V_{n+1}\} \leq \lfloor (n + 1)(1 - \alpha) \rfloor\right) \\ &= \mathbb{P}\left(\text{Unif}\{0, 1, \dots, n - 1, n\} \leq \lfloor (n + 1)(1 - \alpha) \rfloor\right) \\ &= \frac{1 + \lfloor (n + 1)(1 - \alpha) \rfloor}{n + 1} \\ &\geq 1 - \alpha. \end{aligned}$$

Above the step where $\sum_{i=1}^n I\{V_i \leq V_{n+1}\} \sim \text{Unif}\{0, 1, \dots, n - 1, n\}$ follows from exchangeability of $(V_i)_{i=1}^{n+1}$ (and the assumption almost surely $V_i \neq V_j$ for $i \neq j$).

Note you can also cite Lemma's or Theorem's from Lecture 17 in Stats 300C to solve this problem.

- (d) If X_{n+1} is far outside the range of the training data, I would be concerned that the assumption that $(X_1, Y_1), \dots, (X_n, Y_n), (X_{n+1}, Y_{n+1})$ are exchangeable from some distribution P is violated and that the intervals from part (c) are no longer valid. Even if X_{n+1} was technically a draw from P , the collaborator is confusing conditional coverage with marginal coverage. We do not have conditional coverage over all possible values of X_{n+1} . It is very believable that our conditional coverage decreases as X_{n+1} goes to the tails of the distribution of X .

Despite failure to meet the exchangeability assumption, if we went ahead and constructed the prediction intervals defined in (c), we would get prediction intervals with undesirable behavior. In particular, if the curve $\hat{\mu}$ is fit based on

kernel smoothing or local linear regression (and only considers points with similar X values), then it would follow that for all y , $\hat{\mu}_y(X_{n+1}) = y$ implying that $R_{y,n+1} = 0$ for all y , further implying that $\pi(y) = 0$ for all y . Therefore, if $\hat{\mu}$ is fit based on kernel smoothing or local linear regression, the prediction interval would have infinite length. If on the other hand the curve $\hat{\mu}$ is fit based on a global polynomial regression, one would expect $R_{y,n+1}$ to be much larger than $R_{y,i}$ ($i < n$) for most y values in which case the prediction interval would be very small. However, for a global model, since the global model is unlikely to hold for outliers we would want the prediction intervals to be very large. In summary, if we were to fit a curve with large extrapolation bias, the interval from part (c) would be very small and not reflect the extrapolation bias, but if we were to fit a local smooth model, the intervals from part (c) would be infinite length. The collaborator should therefore expect meaningless intervals if they went ahead and used prediction intervals for their outlier point.

Problem 4: PCA versus k-means

- (a) **Explanation for PCA:** Suppose that each of the features is centered such that columns of X each have mean 0. PCA can be thought of in terms of matrix factorizing X . In particular to implement PCA, you take the SVD of X , which is a matrix factorization given by $X = UDV^\top$ where $U \in \mathbb{R}^{n \times n}$ and $V \in \mathbb{R}^{p \times p}$ are orthogonal matrices, and $D_{ij} = 0$ for all $i \neq j$ and $|D_{11}| \geq |D_{22}| \geq \dots \geq 0$. In PCA, the principal component directions are given by the columns of V , while the principal component scores are given by UD . Since $X = (UD)V^\top$, principal component analysis can be thought of as factorizing X into matrix of the principal component scores (given by UD) and a matrix whose rows are the principal component directions (given by V^\top).

Explanation for K-means: K-means can be thought of as an approximate matrix factorization of $X \in \mathbb{R}^{n \times p}$. In particular let

$$\mathcal{C}_K \equiv \left\{ C \in \{0, 1\}^{n \times K} : \sum_{j=1}^K C_{ij} = 1 \text{ for } i = 1, \dots, n \right\},$$

be the collection of $n \times K$ matrices whose rows contain exactly $K - 1$ zeros and 1 one. We can then think of K-means as solving the following approximate matrix factorization problem

$$(\hat{C}, \hat{Z}) = \underset{C \in \mathcal{C}_K, Z \in \mathbb{R}^{K \times p}}{\operatorname{argmin}} \|X - CZ\|_F^2.$$

In particular, if you solve the above approximate matrix factorization problem, the K rows of \hat{Z} will give the K cluster centroids for K-means, and the column index of the nonzero entry in each of the n rows of \hat{C} will give the cluster assignment for each of the n points in the K-means algorithm (or put another way, \hat{C}_{ij}

is a indicator of whether the i th point is assigned to cluster j in the K-means algorithm).

- (b) For PCA, you can determine the matrix V , whose columns give the principal components directions using X_{train} . Then, you can use the principal component scores on your test set given by $X_{\text{test}}V$ as features (or some subset of the principal component scores as features) for predicting Y_{test} . Note that if your prediction method is linear regression, and only the first k principal component scores are used as features, this approach would be principal components regression.

For K -means, you use the training data X_{train} , to determine the K cluster centroids. Once you have the cluster centroids, you can determine the cluster assignment of each observation in the test set by determining which of the K cluster centroids is closest (in Euclidean distance) to each row of the matrix X_{test} . Once you have the cluster assignments on the test set, you can use the cluster assignments as a categorical feature for prediction Y_{test} (you may want to use other features in addition to the cluster assignment).

You can evaluate the error in predicting Y for your choice of prediction algorithm and your choice features using cross-validation on the set $(X_{\text{test}}, Y_{\text{test}})$.

- (c) While my peer isn't wrong that the cluster structure they found is predictive of Y , it is just not such an impressive predictor of Y . In particular my peer's cluster assignments give a statistically significant improvement over just using the grand mean of Y for predicting Y . That being said, looking at the sum of squares in the ANOVA table, using the cluster means to predict Y doesn't give an impressive improvement over using the grand mean to predict Y . In a similar vein, from linear regression on the cluster we also see that the residuals from the linear regression tend to be much larger in absolute value than the estimates of Y for each cluster. So, I agree with my peer that the cluster structure found is predictive of Y , but it doesn't appear to be terribly important structure for predicting Y .
- (d) The results don't contradict each other so it does not cause me to doubt my peer's results. It is possible that the first first 5 principle components are simply not that associated (in a linear way) with the outcome variable Y . Perhaps the j th principal component for $j > 5$ is important in predicting Y . Perhaps the 1st principal component is important in explaining Y , but it has U shaped quadratic relationship with Y with a linear coefficient of 0. In either case the 3-means approach that my friend did would give a better prediction of Y than the principal components regression approach (with 5 components) that I took.

While it doesn't sound like my peer did anything wrong, I'm not terribly impressed by his results because the structure he found is quite a weak predictor of Y (admittedly, I am even less impressed by my own results). I would suggest to our supervisor that we continue seeking an alternative to the 3-means approach,

as there likely is a better choice of features out there. Perhaps it would be an 8-means approach or involve more than the first 5 principal components or it would involve interaction terms. Also neither of our approaches leveraged the Y_{train} data, even though we are told that it exists. I would therefore try to convince our supervisor that it is worth digging deeper before sticking to the 3-means approach of my peer.

Problem 5: Testing and inference on a censored Gaussian draw

Key Ideas/ Main Tools: Score test, maximum likelihood estimation, Bayesian inference

- (a) Z is distributed as $N(\mu, 1)$ variable constrained to be at least 2. Therefore, letting Φ denote the standard Gaussian CDF and ϕ denote the standard Gaussian pdf the likelihood is given by

$$L(\mu) = \frac{\frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}(Z - \mu)^2\right)}{\int_2^\infty \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}(z - \mu)^2\right) dz} = \frac{\phi(Z - \mu)}{1 - \Phi(2 - \mu)} = \frac{\phi(Z - \mu)}{\Phi(\mu - 2)}.$$

The log-likelihood is thus given by

$$l(\mu) = \log(L(\mu)) = \log\left(\frac{1}{\sqrt{2\pi}}\right) - \frac{1}{2}(Z - \mu)^2 - \log(\Phi(\mu - 2)).$$

The score function is therefore given by

$$U(\mu) = l'(\mu) = (Z - \mu) - \frac{\phi(\mu - 2)}{\Phi(\mu - 2)},$$

and the Fisher information is given by

$$\begin{aligned} I(\mu) &= -\mathbb{E}[l''(\mu) \mid \mu] \\ &= -\mathbb{E}\left[-1 - \frac{\Phi(\mu - 2)\phi'(\mu - 2) - [\phi(\mu - 2)]^2}{[\Phi(\mu - 2)]^2} \mid \mu\right] \\ &= 1 + \frac{(2 - \mu)\Phi(\mu - 2)\phi(\mu - 2) - [\phi(\mu - 2)]^2}{[\Phi(\mu - 2)]^2}. \end{aligned}$$

To conduct a score test of the hypothesis $H_0 : \mu = 0$, one would use the test statistic,

$$T = \frac{[U(0)]^2}{I(0)} = \frac{\left(Z - \frac{\phi(-2)}{\Phi(-2)}\right)^2}{1 + \frac{2\Phi(-2)\phi(-2) - [\phi(-2)]^2}{[\Phi(-2)]^2}} = \frac{(Z - r)^2}{1 + 2r - r^2}$$

where $r = \frac{\phi(-2)}{\Phi(-2)} \approx 2.373$. The score test will reject whenever T exceeds $c_{1-\alpha}$, where $c_{1-\alpha}$ is the $1 - \alpha$ quantile of a chi-squared distribution with one degree of

freedom chosen to satisfy $\mathbb{P}(\chi_1^2 \leq c_{1-\alpha}) = 1 - \alpha$. (Note $c_{1-\alpha} \approx 3.84$ for $\alpha = 0.05$). Thus the score test rejects whenever $T > c_{1-\alpha}$, or equivalently

$$\frac{(Z - r)^2}{1 + 2r - r^2} > c_{1-\alpha} \Leftrightarrow Z > r + \sqrt{(1 + 2r - r^2)c_{1-\alpha}} \text{ or } Z < r - \sqrt{(1 + 2r - r^2)c_{1-\alpha}}.$$

Since $Z < 2$ cannot be observed, at level $\alpha = 0.05$, noting that $c_{0.95} \approx 3.84$, the score test of H_0 rejects whenever

$$Z > r + \sqrt{(1 + 2r - r^2)c_{1-\alpha}} \approx 3.036.$$

- (b) Since we just have to optimize over a 1-dimensional parameter you can use grid search to find the MLE. While the following argument is probably unnecessary for the applied qual, to double check that grid search can be used we will find $a, b \in \mathbb{R}$ such that we know $\operatorname{argmax}_{\mu \in \mathbb{R}} l(\mu) \in [a, b]$. Note that whenever, $\mu > Z$, $l'(\mu) < 0$, so the MLE must be at most Z , and we can set $b = Z$. To find a lower endpoint a for the grid search observe that as a consequence of Theorem 1.2.3 in Durrett, for $x < -1$, $\frac{\phi(x)}{\Phi(x)} \leq \left(\frac{1}{-x} - \frac{1}{-x^3}\right)^{-1}$

and hence for $\mu < 1$,

$$\begin{aligned} l'(\mu) &= Z - \mu - \frac{\phi(\mu - 2)}{\Phi(\mu - 2)} \\ &\geq Z - \mu - \left(\frac{1}{-(\mu - 2)} - \frac{1}{-(\mu - 2)^3} \right)^{-1} \\ &= Z - \mu - \frac{(2 - \mu)^3}{(2 - \mu)^2 - 1} \\ &= Z - \mu - \frac{(2 - \mu)^2}{(3 - \mu)(1 - \mu)}(2 - \mu) \\ &= Z - 2 \frac{(2 - \mu)^2}{(3 - \mu)(1 - \mu)} + \frac{\mu(2 - \mu)^2 - \mu(3 - \mu)(1 - \mu)}{(3 - \mu)(1 - \mu)} \\ &= Z - 2 \frac{(2 - \mu)^2}{(3 - \mu)(1 - \mu)} + \frac{\mu}{(3 - \mu)(1 - \mu)}. \end{aligned}$$

Since the above inequality holds for any $\mu < 1$, it is easy to see that $\liminf_{\mu \downarrow -\infty} l'(\mu) \geq Z - 2 > 0$. Further we can use the lower bound above to find an a such that for all $\mu < a$,

$$l'(\mu) \geq Z - 2 \frac{(2 - \mu)^2}{(3 - \mu)(1 - \mu)} + \frac{\mu}{(3 - \mu)(1 - \mu)} > 0.$$

Hence we have an interval $[a, b]$ for which $l'(\mu) > 0$ when $\mu < a$ and $l'(\mu) < 0$ when $\mu > b$. It follows that the MLE $\operatorname{argmax}_{\mu \in \mathbb{R}} l(\mu)$ must lie in $[a, b]$. Hence we can simply perform grid search over $[a, b]$ to find the MLE.

- c) Suppose $\mu \sim \pi$ and an $Z \mid \mu \sim N(\mu, 1)$. We can estimate μ by considering the posterior distribution of μ given Z . In particular, if we observe some $Z > 2$, by Bayes' rule

$$p(\mu \mid Z, Z > 2) = p(\mu \mid Z) \propto \pi(\mu) \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}(Z-\mu)^2\right) \propto \pi(\mu) \exp\left(-\frac{1}{2}(Z-\mu)^2\right).$$

Since the posterior distribution of μ is proportional to $\pi(\mu) \exp\left(-\frac{1}{2}(Z-\mu)^2\right)$, we can estimate μ with the MAP estimate given by

$$\hat{\mu}_{\text{MAP}} = \underset{\mu \in \mathbb{R}}{\operatorname{argmax}} \left\{ \pi(\mu) \exp\left(-\frac{1}{2}(Z-\mu)^2\right) \right\}.$$

As in part (b) this estimator can be found using 1-dimensional grid search. An alternative estimate for μ would be to estimate the posterior mean

$$\hat{\mu} = \mathbb{E}[\mu \mid Z] = \frac{\int_{-\infty}^{\infty} \mu \pi(\mu) \exp\left(-\frac{1}{2}(Z-\mu)^2\right) d\mu}{\int_{-\infty}^{\infty} \pi(\mu) \exp\left(-\frac{1}{2}(Z-\mu)^2\right) d\mu}.$$

The numerator and denominator of the above expression can each be approximated numerically using a Gauss–Hermite quadrature. Alternatively, the above posterior mean can be estimated using the Metropolis–Hastings algorithm.

- (c) I'm not entirely sure what the question means by “conflict”, but the answer in part (c) was a Bayesian approach based on one observation for which $Z > 2$, whereas part (a) and part (b) describe frequentist approaches where Z is drawn until $Z > 2$. The answer in part (c) used a different likelihood than was used in parts (a) and (b). In particular the likelihood in items (a) and (b) was

$$p(Z \mid \mu, Z > 2) = \phi(Z - \mu) / \Phi(\mu - 2),$$

whereas the likelihood in part (c) that was used was

$$p(Z \mid \mu) = \phi(Z - \mu).$$

The easiest way to see why it was not a mistake to use a different likelihood in part (c), is to note that we could have used the same likelihood in part (c) when applying Bayes' rule as the likelihood used in (a) and (b), but doing so would have made a more difficult calculation. In particular, conditioning on $Z > 2$, we could have used Bayes' rule as follows

$$p(\mu \mid Z, Z > 2) = \frac{p(\mu \mid Z > 2)p(Z \mid \mu, Z > 2)}{\int_{-\infty}^{\infty} p(\mu \mid Z > 2)p(Z \mid \mu, Z > 2)d\mu} \propto p(\mu \mid Z > 2)p(Z \mid \mu, Z > 2),$$

and used the likelihood from parts (a) and (b), but $\pi(\mu \mid Z > 2)$ is more difficult to work with than $\pi(\mu)$ and requires applying Bayes' rule. In fact, if we apply Bayes' rule to $\pi(\mu \mid Z > 2)$ in the above expression, we simply recover the approach used in part (c):

$$p(\mu \mid Z, Z > 2) \propto \pi(\mu)p(Z > 2 \mid \mu)p(Z \mid \mu, Z > 2) = \pi(\mu)\Phi(\mu-2)\frac{\phi(Z-\mu)}{\Phi(\mu-2)} = \pi(\mu)\phi(Z-\mu).$$

Problem 6: Cross-validation in the normal linear model

Key Ideas/ Main Tools: Cross validation, properties of the normal linear model.

This problem is based on results from Bates et al. [2022]. Note that the 2021 qual was open internet, so I think that those who were aware of the paper or were able to find the paper found it the problem quite straightforward, but those who didn't found part (b) especially tricky.

- (a) Fix any $x_1, x_2, \dots, x_n, y_1, \dots, y_n, \kappa$ and u . Now for any subset $S \subset [n]$, let $\hat{\theta}_{S,0}$ denote the OLS estimator trained on the points $\{(x_i, y_i)\}_{i \in S}$ and let $\hat{\theta}_{S,\kappa}$ denote the OLS estimator trained on the shifted points $\{(x_i, y_i + x_i^\top \kappa)\}_{i \in S}$. Also let $\mathcal{X}_S \in \mathbb{R}^{|S| \times p}$ be the design matrix for these OLS regressions whose rows consist of $\{x_i : i \in S\}$ and letting $\mathcal{Y}_S = (y_i)_{i \in S}$ be the vector of outcomes for the OLS regression to obtain $\hat{\theta}_{S,0}$. Observe that by the formula for an OLS estimator

$$\hat{\theta}_{S,\kappa} = (\mathcal{X}_S^\top \mathcal{X}_S)^{-1} \mathcal{X}_S^\top [\mathcal{Y}_S + \mathcal{X}_S \kappa] = (\mathcal{X}_S^\top \mathcal{X}_S)^{-1} \mathcal{X}_S^\top \mathcal{Y}_S + \kappa = \hat{\theta}_{S,0} + \kappa.$$

For any $j \notin S$, if we let $\hat{y}_{S,0,j} = x_j^\top \hat{\theta}_{S,0}$ be the OLS prediction at point j when training on the subset S for the raw data (x_i, y_i) and if we let $\hat{y}_{S,\kappa,j} = x_j^\top \hat{\theta}_{S,\kappa}$ be the OLS prediction at point j when training on the subset S of the translated data $(x_i, y_i + x_i^\top \kappa)$, it follows that

$$\begin{aligned} \ell(\hat{y}_{S,\kappa,j}, y_j + x_j^\top \kappa) &= \ell(x_j^\top \hat{\theta}_{S,\kappa}, y_j + x_j^\top \kappa) \\ &= \ell(x_j^\top \hat{\theta}_{S,0} + x_j^\top \kappa, y_j + x_j^\top \kappa) \\ &= \ell(x_j^\top \hat{\theta}_{S,0}, y_j) \\ &= \ell(\hat{y}_{S,0,j}, y_j), \end{aligned}$$

where the 2nd last steps holds because ℓ is the squared error loss function. It is clear that above argument holds for any subset S and $j \notin S$.

Letting $S_1(u), \dots, S_K(u)$ be the K training subsets of $[n]$ that define the cross-validation (which depend on the random draw U which we are fixing to be u), note that applying the previous result,

$$\begin{aligned} \widehat{\text{Err}}^{(\text{CV})}((x_1, y_1), \dots, (x_n, y_n), u) &\equiv \frac{1}{n} \sum_{k=1}^K \sum_{j \notin S_k(u)} \ell(\hat{y}_{S_k(u),0,j}, y_j) \\ &= \frac{1}{n} \sum_{k=1}^K \sum_{j \notin S_k(u)} \ell(\hat{y}_{S_k(u),\kappa,j}, y_j + x_j^\top \kappa) \\ &= \widehat{\text{Err}}^{(\text{CV})}((x_1, y_1 + x_1^\top \kappa), \dots, (x_n, y_n + x_n^\top \kappa), u). \end{aligned}$$

Since this argument holds for any fixed $x_1, x_2, \dots, x_n, y_1, \dots, y_n, \kappa$ and u , it follows that $\widehat{\text{Err}}^{(\text{CV})}$ is linearly invariant by definition (2).

- (b) Recalling from the problem statement that $\hat{\theta}$ is the OLS estimator based on the all of the observed data, and let (R_1, \dots, R_n) be the residuals for the OLS estimate on the observed data (i.e. $R_i = Y_i - X_i^\top \hat{\theta}$ for all $i \in [n]$). By part (a), if we let $\kappa = -\hat{\theta}$,

$$\begin{aligned}\widehat{\text{Err}}^{(\text{CV})}\left((X_1, Y_1), \dots, (X_n, Y_n), U\right) &= \widehat{\text{Err}}^{(\text{CV})}\left((X_1, Y_1 - X_1^\top \hat{\theta}), \dots, (X_n, Y_n - X_n^\top \hat{\theta}), U\right) \\ &= \widehat{\text{Err}}^{(\text{CV})}\left((X_1, R_1), \dots, (X_n, R_n), U\right).\end{aligned}$$

It follows conditional on $X = (X_1, \dots, X_n)$, $\widehat{\text{Err}}^{(\text{CV})}$ is a function of only the residuals $(R_1, R_2, \dots, R_n, U)$. Also observe that conditional on X , Err_{XY} is only a function of $\hat{\theta}$. By a property of linear regression under the homoskedastic linear model, $(R_1, R_2, \dots, R_n) \perp\!\!\!\perp \hat{\theta} | X$ (to see this, one can check using the hat matrix that conditional on X , the residuals are uncorrelated with the estimator $\hat{\theta}$ and note that for multivariate Gaussian's zero correlation implies independence). Since U is independent of the data, this further implies that

$$(R_1, R_2, \dots, R_n, U) \perp\!\!\!\perp \hat{\theta} | X.$$

Because conditional on X , we have shown that $\widehat{\text{Err}}^{(\text{CV})}$ is only a function of $(R_1, R_2, \dots, R_n, U)$ and because conditional on X , Err_{XY} is only a function of $\hat{\theta}$ the conditional independence result displayed above implies that

$$\widehat{\text{Err}}^{(\text{CV})} \perp\!\!\!\perp \text{Err}_{XY} | X.$$

- c) The previous result from item (b) does not imply that $\widehat{\text{Err}}^{(\text{CV})}$ is a useless estimate of prediction error. In particular, $\widehat{\text{Err}}^{(\text{CV})}$ is still a good estimate of $\text{Err} \equiv \mathbb{E}[\text{Err}_{XY}]$, which is the expected prediction loss across all training sets (see Chapter 7.12 in Hastie et al. [2009] and Bates et al. [2022]). Even if we are truly interested in estimating Err_{XY} rather than Err , just because $\widehat{\text{Err}}^{(\text{CV})}$ is uncorrelated with Err_{XY} , it does not mean that it is a bad approximation of Err_{XY} : for large n , the random variable Err_{XY} will likely be concentrated closely about its mean $\text{Err} = \mathbb{E}[\text{Err}_{XY}]$.
- (c) Sample splitting into a training set and a test set would also give a linearly invariant estimate of the prediction error by a similar argument in part (a) and the same argument in part (b). Another commonly used estimate of prediction error that is linearly invariant, so that (3) holds is Mallows's C_p . See Bates et al. [2022] for discussion about Mallows's C_p and other commonly used linearly invariant estimates of prediction error.

11 Applied 2022¹⁷

Problem 1: R-squared and PCA

Key ideas:

- Connections between principal components analysis and the singular value decomposition.
 - Orthogonality of principal components and principal component directions.
- (a) We are given that $\mathbf{X} \in \mathbb{R}^{n \times p}$ is standardized so that the columns have mean zero and variance one. We are also given that $\mathbf{X} = \mathbf{U}\mathbf{D}\mathbf{V}^\top$ is the SVD of \mathbf{X} . Let $d_1 \geq d_2 \geq \dots \geq d_p \geq 0$ be the diagonal entries of \mathbf{D} . The columns of \mathbf{V} are thus the principal component directions and $\frac{1}{n}d_j^2 = \frac{1}{n}\|\mathbf{X}\mathbf{v}_j\|_2^2$ is the variance of \mathbf{X} in the j th principal component direction. The cumulative percent variance explained sequence is thus,

$$\rho_k = 100 \times \frac{\sum_{s=1}^k \frac{1}{n}d_s^2}{\sum_{s=1}^p \frac{1}{n}d_s^2} = 100 \times \frac{\sum_{s=1}^k d_s^2}{\sum_{s=1}^p d_s^2},$$

for $k = 1, \dots, p$.

- (b) We are given a response $\mathbf{y} \in \mathbb{R}^n$ with mean zero and variance one. The fitted values are $\hat{\mathbf{y}} = \mathbf{X}\hat{\boldsymbol{\beta}} \in \mathbb{R}^n$. We know that the fitted values $\hat{\mathbf{y}}$ are orthogonal to the residuals $\mathbf{y} - \hat{\mathbf{y}}$. Thus,

$$\|\mathbf{y}\|_2^2 = \|\mathbf{y} - \hat{\mathbf{y}} + \hat{\mathbf{y}}\|_2^2 = \|\mathbf{y} - \hat{\mathbf{y}}\|_2^2 + \|\hat{\mathbf{y}}\|_2^2.$$

And so

$$\frac{1}{n} \sum_{i=1}^n y_i^2 = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 + \frac{1}{n} \sum_{i=1}^n \hat{y}_i^2.$$

Since \mathbf{y} has mean zero and variance one, this implies that

$$1 = \text{MSE}_0 = \text{MSE} + \text{MSS}.$$

And so,

$$R^2 = 1 - \frac{\text{MSE}}{\text{MSE}_0} = 1 - \text{MSE} = \text{MSS}.$$

- (c) We now regress the j th column of \mathbf{X} on the first k principal components of \mathbf{X} . Let $\mathbf{U}_k = [\mathbf{u}_1, \dots, \mathbf{u}_k] \in \mathbb{R}^{n \times k}$ be the matrix containing the first k columns of \mathbf{U} . Since $\mathbf{U}_k^\top \mathbf{U}_k = \mathbf{I}_k$, the fitted values from regressing \mathbf{x}_j on \mathbf{U}_k are

$$\hat{\mathbf{x}}_j = \mathbf{U}_k(\mathbf{U}_k^\top \mathbf{U}_k)^{-1} \mathbf{U}_k^\top \mathbf{x}_j = \mathbf{U}_k \mathbf{U}_k^\top \mathbf{x}_j.$$

¹⁷Michael Howes

The average regression sum of squares is thus,

$$\begin{aligned} \text{MSS}_j &= \frac{1}{n} \hat{\mathbf{x}}_j^\top \hat{\mathbf{x}}_j \\ &= \frac{1}{n} \mathbf{x}_j^\top \mathbf{U}_k \mathbf{U}_k^\top \mathbf{U}_k \mathbf{U}_k^\top \mathbf{x}_j \\ &= \frac{1}{n} \mathbf{x}_j^\top \mathbf{U}_k \mathbf{U}_k^\top \mathbf{x}_j. \end{aligned}$$

Since \mathbf{x}_j is the j th column of \mathbf{X} we $\mathbf{x}_j = \mathbf{X} \mathbf{e}_j$ where $\mathbf{e}_j \in \mathbb{R}^p$ is the j th standard basis vector.

$$\begin{aligned} \mathbf{U}_k^\top \mathbf{x}_j &= \mathbf{U}_k^\top \mathbf{X} \mathbf{e}_j \\ &= \mathbf{U}_k^\top \mathbf{U} \mathbf{D} \mathbf{V}^\top \mathbf{e}_j. \end{aligned}$$

We know that $\mathbf{U}_k^\top \mathbf{U} = [\mathbf{I}_k, \mathbf{0}_{k \times (p-k)}] \in \mathbb{R}^{k \times p}$ where $\mathbf{0}_{k \times (p-k)}$ is a matrix of all zeros of size $k \times (p-k)$. It follows that

$$\begin{aligned} \mathbf{U}_k^\top \mathbf{U} \mathbf{D} \mathbf{V}^\top &= [\mathbf{I}_k, \mathbf{0}_{k \times (p-k)}] \mathbf{D} \mathbf{V}^\top \\ &= \mathbf{D}_k \mathbf{V}_k^\top, \end{aligned}$$

where $\mathbf{D}_k = \text{diag}(d_1, \dots, d_k) \in \mathbb{R}^{k \times k}$ and $\mathbf{V}_k \in \mathbb{R}^{p \times k}$ is equal to the first k rows of \mathbf{V} . Thus,

$$\begin{aligned} \mathbf{U}_k^\top \hat{\mathbf{x}}_j &= \mathbf{D}_k \mathbf{V}_k^\top \mathbf{e}_j \\ &= \sum_{s=1}^k d_s (\mathbf{v}_s^\top \mathbf{e}_j) \mathbf{e}_s \\ &= \sum_{s=1}^k d_s v_{sj} \mathbf{e}_s \end{aligned}$$

where v_{sj} is the entry of \mathbf{V} in row s and column j . We thus have

$$\begin{aligned} \text{MSS}_j &= \frac{1}{n} \|\mathbf{U}_k^\top \hat{\mathbf{x}}_j\|_2^2 \\ &= \frac{1}{n} \left\| \sum_{s=1}^k d_s v_{sj} \mathbf{e}_s \right\|_2^2 \\ &= \frac{1}{n} \sum_{s=1}^k d_s^2 v_{sj}^2. \end{aligned}$$

Since \mathbf{x}_j has mean zero and variance one, we are in the setting of part (b) and hence

$$R_j^2 = \text{MSS}_j = \frac{1}{n} \sum_{s=1}^k d_s^2 v_{sj}^2.$$

(d) Note that

$$\begin{aligned}
\sum_{j=1}^p \text{MSS}_j &= \sum_{j=1}^p \frac{1}{n} \sum_{s=1}^k d_s^2 v_{sj}^2 \\
&= \sum_{s=1}^k \sum_{j=1}^p \frac{1}{n} d_s^2 v_{sj}^2 \\
&= \sum_{s=1}^k \frac{1}{n} d_s^2 \sum_{j=1}^p v_{sj}^2 \\
&= \sum_{s=1}^k \frac{1}{n} d_s^2 \|\mathbf{v}_s\|_2^2 \\
&= \frac{1}{n} \sum_{s=1}^k d_s^2,
\end{aligned}$$

since all rows of \mathbf{V} have norm one. We know that each column of \mathbf{X} has variance one and mean zero. Thus,

$$\begin{aligned}
p &= \sum_{j=1}^p \frac{1}{n} \mathbf{x}_j^\top \mathbf{x}_j \\
&= \frac{1}{n} \sum_{j=1}^p \text{tr}(\mathbf{x}_j^\top \mathbf{x}_j) \\
&= \frac{1}{n} \sum_{j=1}^p \text{tr}(\mathbf{x}_j \mathbf{x}_j^\top) \\
&= \frac{1}{n} \text{tr} \left(\sum_{j=1}^p \mathbf{x}_j \mathbf{x}_j^\top \right) \\
&= \frac{1}{n} \text{tr}(\mathbf{X} \mathbf{X}^\top) \\
&= \frac{1}{n} \text{tr}(\mathbf{U} \mathbf{D} \mathbf{V}^\top \mathbf{V} \mathbf{D} \mathbf{U}^\top) \\
&= \frac{1}{n} \text{tr}(\mathbf{D}^2) \\
&= \frac{1}{n} \sum_{s=1}^p d_s^2.
\end{aligned}$$

Thus,

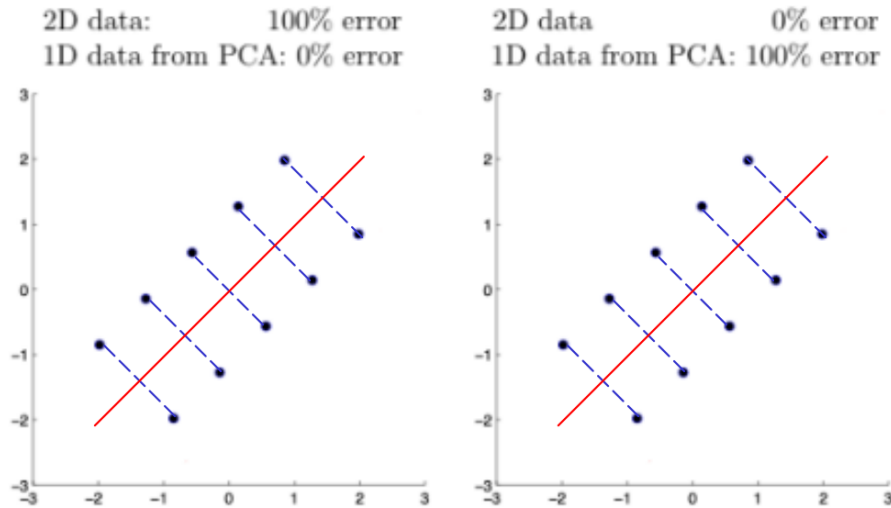
$$\frac{100}{p} \sum_{j=1}^p \text{MSS}_j = \frac{100}{np} \sum_{s=1}^k d_s^2 = 100 \times \frac{\sum_{s=1}^k d_s^2}{\sum_{s=1}^p d_s^2} = \rho_k.$$

So $\frac{100}{p} \sum_{j=1}^p \text{MSS}_j$ is exactly the cumulative percent variance explained sequence.

Problem 2: LOO, PCA and 1NN

Key ideas:

- “Eyeballing” principal component directions.
 - Working out nearest neighbor classifiers.
1. We are asked to draw a line corresponding to the first principal component. This does not have to be exact, but we can see that a roughly 45 degree line gives the direction with the most variance. Here the principal component direction is represented as a red solid line. In both plots I have also included dashed lines showing the projection onto the first principal direction.

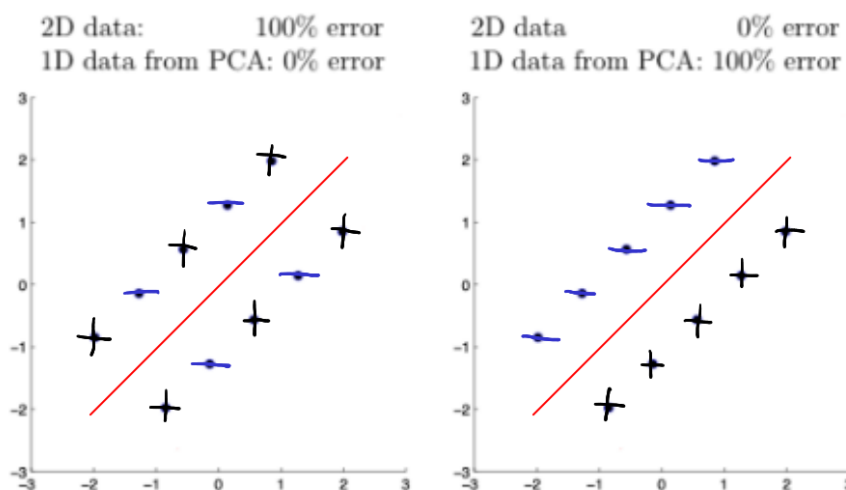


2. Looking at the previous figure we can make two observations.
 - Using the 2D data, the nearest neighbor of a point is one of the adjacent points on the line of points parallel to the PCA direction. For example, for the point at roughly $(-1, -2)$, the nearest neighbor is the point at roughly $(0, -1.25)$.
 - Using the projected 1D data from PCA, the nearest neighbor of a point is the point across the PCA direction. This because these points get projected onto the same value when we perform PCA. For example, for the point at roughly $(-1, -2)$, the 1D projected nearest neighbor is the point at roughly $(-2, -1)$.

To have 100% error on the 2D data we want an alternating sequence of “+”’s and “−”’s along the two lines of points parallel to the PCA direction. If we use the same sequence of “+”’s and “−”’s on both lines of points, then we will also have 0% error when using the 1D data. This is because the 1D-nearest neighbor points will have the same labels.

To have 0% error on the 2D data we want adjacent points on the two lines to all have the same label. If we label only line of points with “+”’s and the other with “−”’s, then the 1D data will also have 100% error. This is because the 1D nearest neighbors will have the opposite labels.

In summary, the labelling below has the specified error rates.



Question 3: Financial Data Science

Key ideas

- General knowledge about supervised learning algorithms.
 - The dangers of using cross-validation on non i.i.d. data.
- (a) We are asked to describe the strengths of four different supervised learning algorithms for predicting stock returns. These are ridge regression, lasso and elastic net, random forests and gradient boosting. The goal here should be to just say something for each method. It doesn’t have to be very formal or precise.
- **Ridge regression** is good at combining many signals across different predictors. Ridge regression can combine weak signals from different predictors to give a strong signal.

- **Lasso and elastic net** is good for feature selection. If we expect that only a few predictors are important for predicting stock returns, then lasso or elastic net would be a good choice since they induce sparsity.
 - **Random forests** are non-linear models, and so they are more flexible than the previous two. Random forests are more interpretable and easier to tune than other non-linear models like neural networks.
 - **Gradient boosting** is also a non-linear model and hence flexible. Gradient boosting is sequential algorithm, and we can use cross-validation to tune the number of steps used.
- (b) We are asked to list any other predictors that would be useful for predicting daily stock returns. Temporal data such as day of the week or month of the year may be helpful. Previous daily stock returns could also be used as a feature. The age of company could also be a useful predictor.

We could also try transforming some of our predictors. For example, we could use log-profits instead of profits or add interactions between sector and company size. It might also be worth-while to transform the response variable. This would be more important if we use a linear model like ridge regression or lasso/elastic net.

- (c) It is not surprising that the test error is much higher than the cross-validation error. This is because our data is temporally correlated. When doing cross-validation we have ignored the temporal dependency. Each data point in the held-out fold will contain points in the other folds that are close in time. On the other hand, the test data contains data from different years to the training data. This means it is much easier to make predictions on the held-out fold than on the test data. As a result, our cross-validation estimate of the test error is biased downwards.

There is also a second, less severe, problem. The cross-validation procedure was used for model selection not to estimate the test error directly. The estimate of test error is again likely to be downward biased for the test error since we have specifically chosen a model with low CV-error.

- (d) We need to use a cross-validation process that respects the temporal dependency. A simple way to do this is by “blocking” our data. We could drop data into groups of 1-year or 6-months. Then we should create our CV-folds by choosing groups of data points instead of individual data points. This will ensure that the held-out folds do not contain data points that are close in time to data points in the training folds. This procedure will lead to choosing a different model which hopefully performs better on the training data.

The suggestion above does not address the second concern and there may again be some downward bias for selection. In this case, it is not particular important.

Since we have a separate test data set, we can just use the test data to estimate the test error from deploying the model. If we want to use all the data from 2010-2019 to train our model and get an estimate of the test error, then we could do nested cross-validation. Again the folds in the nested cross-validation should be based on choosing groups of data points from the same time period not on choosing data points individually.

- (e) The out-of-bag error in a random forest is an estimate of test error that is “almost identical to that obtained by N -fold cross-validation” (Section 15.3 of [Hastie, Tibshirani, and Friedman, 2009]). We would therefore expect the out-of-bag error to be higher than the test error for the same reasons discussed in part (c).

Question 4: Infectious Disease Survival Times

Key ideas:

- Testing coefficients in the proportional hazard model.
 - Testing independence in $2 \times 2 \times K$ tables with the Mantel–Haenszel test.
- (a) Since our data is right censored, we will use tools from survival analysis. Since we want to control for affects from each region, we can use a proportional hazard model. Suppose we have N subjects and for each subject $i = 1, \dots, N$ we observe
- An observed time $O_i = \min\{C_i, T_i\}$. The time T_i is the event time (in this case the time at which the subject died after infect). The time C_i is the censoring time which we assume is independent of T_i .
 - A censor-indicator $\delta_i = I_{\{O_i=T_i\}} \in \{0, 1\}$ with $\delta_i = 0$ meaning subject i is censored.
 - The vaccination status $V_i \in \{0, 1\}$, we’ll assume that $V_i = 1$ means the subject is vaccinated.
 - The region $R_i \in \{1, \dots, J\}$ where J is the number of regions (assumed to be small).

A proportional hazard models that include effects from both vaccination status and subject regions is

$$T_i \mid V_i, R_i \stackrel{\text{ind}}{\sim} h_i(t), \quad h_i(t) = h_0(t) \exp(\alpha V_i + \beta_{R_i}), \quad (48)$$

where we are modelling the distribution of T_i in terms of T_i ’s hazard function $h_i(t)$. To make this model identifiable, we must add a constraint such as

$$\sum_{j=1}^J \beta_j = 0 \quad \text{or} \quad \beta_J = 0.$$

If the number of regions J is not too large, then we will be able to fit the above model by maximizing the partial likelihood. If the number of regions is large, but we have measured features X_i for subjects i 's region, then we could use the model

$$T_i \mid V_i, R_i \stackrel{\text{ind}}{\sim} h_i(t), \quad h_i(t) = h_0(t) \exp(\alpha V_i + \gamma^\top X_i).$$

Assuming the dimension of X_i is less than $J - 1$, then fitting this second model would be easier than fitting (48). We are told that there are only a “handful” of possible value of R , thus we will assume we have enough data to fit (48). To test the effectiveness of the vaccine, we can test the null hypothesis $\alpha = 0$. If we get evidence that $\alpha < 0$, then we can conclude that vaccination likely has a positive effect after controlling for regions.

An alternative model would be to include an interaction affect between region and vaccination status. This model would be appropriate if we expect the effectiveness of the vaccine to vary across regions. The corresponding proportional hazard model is,

$$T_i \mid V_i, R_i \stackrel{\text{ind}}{\sim} h_i(t), \quad h_i(t) = h_0(t) \exp(\alpha_{R_i} V_i + \beta_{R_i}).$$

Again, testing $\alpha_j = 0$ across the regions j will test for the vaccine's effectiveness in region j . This probably is not a good choice of model. Firstly, we might not have enough data to fit both α_j and β_j . Secondly, while it is reasonable to expect health outcomes to vary across regions (represented by β_j) it is unlikely that the effectiveness of the vaccine will vary across regions (represented by α_j). We will thus stick with model (48)

- (b) In this question, we will ignore the variation across regions. The corresponding Cox proportional hazards model is

$$T_i \mid V_i, R_i \stackrel{\text{ind}}{\sim} h_i(t), \quad h_i(t) = h_0(t) \exp(\alpha V_i)$$

This model is fit by maximizing the log partial likelihood,

$$\ell(\alpha) = \sum_{i: \delta_i=1} \alpha V_i - \log \left(\sum_{k \in R(O_i)} \exp(\alpha V_k) \right), \quad (49)$$

where $R(O_i) = \{k : O_k \geq O_i\}$ is the risk-set at time O_i . For the null $H_0 : \alpha = 0$, the score-test statistic is

$$T = \frac{\ell'(0)^2}{-\ell''(0)},$$

which has asymptotic distribution χ_1^2 . We know that the score-test in a Cox proportional hazards model is equivalent to the log-rank test. The log-rank test is itself a special case of the Mantel–Haenszel test. For each i such that $\delta_i = 1$, we can make the following contingency table based on the risk set at time O_i .

	$V_k = 0$	$V_k = 1$
$O_k = O_i$ and $\delta_k = 1$	Number of unvaccinated subjects that died at time O_i	Number of vaccinated subjects that died at time O_i
$O_k > O_i$ or $\delta_k = 0$	Number of unvaccinated subjects that survived beyond time O_i	Number of vaccinated subjects that survived beyond time O_i

Specifically, we consider all individuals with $O_k \geq O_i$ and put them into one of the four below categories. The null hypothesis that vaccination does not affect survival time implies that the row variable and column variable in the above 2×2 contingency table are equivalent. If there was just one such table, then we could condition on the row and column sums and use Fisher exact test. However, we actually have $K = |\{i : \delta_i = 1\}|$ such tables, and we need a way of combining them. The Mantel–Haenszel test uses the hypergeometric distribution from Fisher’s exact test to combine these tables into a test statistic that is asymptotically distributed according to χ_1^2 . It turns out that this statistic is exactly the score statistic T . This can be proved using the formula for the partial likelihood (49) and the formula for the Mantel–Haenszel test.

- (c) Suppose we are now using model (48) which allows variation across regions. If we wish to do inference on the affect of vaccination, we can still use the score test. The two hypothesis we are comparing are now,

$$H_0 : \alpha = 0 \quad \text{vs} \quad H_1 : \alpha \text{ unconstrained.}$$

Thus, we wish to test if the sub-model which does not include vaccination status is sufficient to explain our data. To make both the null and alternative models identifiable we will add the constraint $\beta_J = 0$ and think of β as a vector in \mathbb{R}^{J-1} . The log partial likelihood is equal to

$$\ell(\alpha, \beta) = \sum_{i:\delta_i=0} \alpha V_i + \beta_{R_i} - \log \left(\sum_{k \in R(O_i)} \exp(\alpha V_k + \beta_{R_k}) \right).$$

Let

$$\begin{aligned} U(\alpha, \beta) &= \nabla_{\alpha, \beta} \ell(\alpha, \beta) \in \mathbb{R}^J, \\ I(\alpha, \beta) &= -\nabla_{\alpha, \beta}^2 \ell(\alpha, \beta) \in \mathbb{R}^{J \times J}, \end{aligned}$$

be the gradient and negative Hessian of ℓ evaluated at (α, β) . Let $\hat{\beta}_0 \in \mathbb{R}^{J-1}$ be the MLE of the Cox proportional hazards model under the constraints $\alpha = 0$ and $\beta_J = 0$. The score statistic for testing H_0 is

$$T = U(0, \hat{\beta}_0)^\top I(0, \hat{\beta}_0)^{-1} U(0, \hat{\beta}_0) \sim \chi_1^2, \quad (50)$$

where the approximation above is asymptotic under the null. As in part (b), we could again use contingency tables to test H_0 . Again we have a 2×2 table for uncensored observation i . However, now our table should only include people in the same region of subject i . Specifically, if $\delta_i = 1$ for some i , then we make the following table This is the same as the contingency table in part (b)

	$V_k = 0$ and $R_k = R_i$	$V_k = 1$ and $R_k = R_i$
$O_k = O_i$ and $\delta_k = 1$	Number of unvaccinated subjects that died at time O_i	Number of vaccinated subjects that died at time O_i
$O_k > O_i$ or $\delta_k = 0$	Number of unvaccinated subjects that survived beyond time O_i	Number of vaccinated subjects that survived beyond time O_i

but now we only included individuals in the same region as subject i . Under the null hypothesis $\alpha = 0$, the row and column variables are independent. Thus, conditional on the row and column, the value in the top square is hypergeometric as in Fisher's exact test. We can thus combine these table via the Mantel–Haenszel test and get a test statistic which will be asymptotically χ_1^2 under the null. However, I do not believe that this is the same test statistic as (50).

When adding the region covariates, we have two ways of testing $\alpha = 0$. One is based on the score test and the other is a variant of the Mantel–Haenszel test. While these tests are likely to be related, I believe that they are different.

Question 5: EM for a mixture of Student-t distributions

Key ideas

- Latent variable models.
- The EM algorithm.

(a) Consider the following augmented model. Independently for $1 \leq n \leq N$,

$$\begin{aligned}
 z_n &\sim \boldsymbol{\pi}, \\
 \tau_n \mid z_n &\sim \text{Ga}(\nu_{z_n}/2, \nu_{z_n}/2), \\
 \mathbf{x}_n \mid \tau_n, z_n &\sim \mathcal{N}(\boldsymbol{\mu}_{z_n}, \tau_n^{-1} \boldsymbol{\Sigma}_{z_n}),
 \end{aligned}$$

the joint density for $\{z_n, \tau_n, \mathbf{x}_n\}_{n=1}^N$ corresponding to this model is

$$\begin{aligned}
& p(\{z_n, \tau_n, \mathbf{x}_n\}_{n=1}^N; \boldsymbol{\theta}) \\
&= \prod_{n=1}^N p(z_n, \tau_n, \mathbf{x}_n; \boldsymbol{\theta}) \\
&= \prod_{n=1}^N p(z_n; \boldsymbol{\theta}) p(\tau_n \mid z_n; \boldsymbol{\theta}) p(\mathbf{x}_n \mid \tau_n, z_n; \boldsymbol{\theta}) \\
&= \prod_{n=1}^N \pi_{z_n} \text{Ga}(\tau_n; \nu_{z_n}/2, \nu_{z_n}/2) \mathcal{N}(\mathbf{x}_n; \boldsymbol{\mu}_{z_n}, \tau_n^{-1} \boldsymbol{\Sigma}_{z_n}).
\end{aligned}$$

The marginal model for $\{z_n, \mathbf{x}_n\}_{n=1}^N$ is

$$\begin{aligned}
& p(\{z_n, \mathbf{x}_n\}; \boldsymbol{\theta}) \\
&= \int_{\mathbb{R}_{>0}^N} p(\{z_n, \tau_n, \mathbf{x}_n\}; \boldsymbol{\theta}) d\boldsymbol{\tau} \\
&= \int_{\mathbb{R}_{>0}^N} \prod_{n=1}^N \pi_{z_n} \text{Ga}(\tau_n; \nu_{z_n}/2, \nu_{z_n}/2) \mathcal{N}(\mathbf{x}_n; \boldsymbol{\mu}_{z_n}, \tau_n^{-1} \boldsymbol{\Sigma}_{z_n}) d\boldsymbol{\tau} \\
&= \prod_{n=1}^N \int \pi_{z_n} \text{Ga}(\tau_n; \nu_{z_n}/2, \nu_{z_n}/2) \mathcal{N}(\mathbf{x}_n; \boldsymbol{\mu}_{z_n}, \tau_n^{-1} \boldsymbol{\Sigma}_{z_n}) d\tau_n \\
&= \prod_{n=1}^N \pi_{z_n} \int \text{Ga}(\tau_n; \nu_{z_n}/2, \nu_{z_n}/2) \mathcal{N}(\mathbf{x}_n; \boldsymbol{\mu}_{z_n}, \tau_n^{-1} \boldsymbol{\Sigma}_{z_n}) d\tau_n \\
&= \prod_{n=1}^N \pi_{z_n} \text{St}(\mathbf{x}_n; \nu_{z_n}, \boldsymbol{\mu}_{z_n}, \tau_n^{-1} \boldsymbol{\Sigma}_{z_n}),
\end{aligned}$$

where the last line follows from the equation describing the Student-t distribution as a scale mixture of Gaussian. This joint probability corresponds to the original student mixture model.

(b) We are asked to compute,

$$\omega_{nk} = p(z_n = k \mid \mathbf{x}_n; \boldsymbol{\theta}),$$

for $1 \leq k \leq K$ and $1 \leq n \leq N$. By Bayes rule,

$$\begin{aligned}
\omega_{nk} &= \frac{p(\mathbf{x}_n \mid z_n = k; \boldsymbol{\theta}) p(z_n = k; \boldsymbol{\theta})}{\sum_{j=1}^K p(\mathbf{x}_n \mid z_n = j; \boldsymbol{\theta}) p(z_n = j; \boldsymbol{\theta})} \\
&= \frac{\text{St}(\mathbf{x}_n; \nu_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \pi_k}{\sum_{j=1}^K \text{St}(\mathbf{x}_n; \nu_j, \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j) \pi_j},
\end{aligned}$$

since $\mathbf{x}_n \mid z_n \sim \text{St}(\nu_{z_n}, \boldsymbol{\mu}_{z_n}, \boldsymbol{\Sigma}_{z_n})$.

(c) Next we are asked to compute the conditional distribution,

$$p(\tau_n \mid z_n = k, \mathbf{x}_n; \boldsymbol{\theta}).$$

To do this, we will again use Bayes rule conditional on $z_n = k$,

$$\begin{aligned} & p(\tau_n \mid z_n = k, \mathbf{x}_n; \boldsymbol{\theta}) \\ & \propto p(\tau_n \mid z_n = k; \boldsymbol{\theta}) p(\mathbf{x}_n \mid \tau_n, z_n = k; \boldsymbol{\theta}) \\ & = \text{Ga}(\tau_n; \nu_k/2, \nu_k/2) \mathcal{N}(\mathbf{x}_n; \boldsymbol{\mu}_k, \tau_n^{-1} \boldsymbol{\Sigma}_k) \\ & \propto \tau_n^{\nu_k/2-1} \exp(-\nu_k \tau/2) \det(\tau_n^{-1} \boldsymbol{\Sigma}_k)^{-1/2} \exp\left(-\frac{1}{2}(\mathbf{x}_n - \boldsymbol{\mu}_k)^\top (\tau_n^{-1} \boldsymbol{\Sigma}_k)^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_k)\right). \end{aligned}$$

Note that,

$$\det(\tau_n^{-1} \boldsymbol{\Sigma}_k) = \det(\tau_n^{-1} \mathbf{I}_D \boldsymbol{\Sigma}_k^{-1}) = \tau_n^{-D} \det(\boldsymbol{\Sigma}_k^{-1}),$$

and

$$-\frac{1}{2}(\mathbf{x}_n - \boldsymbol{\mu}_k)^\top (\tau_n^{-1} \boldsymbol{\Sigma}_k)^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_k) = -\frac{1}{2}(\mathbf{x}_n - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_k) \tau_n.$$

Thus,

$$\begin{aligned} & p(\tau_n \mid z_n = k, \mathbf{x}_n; \boldsymbol{\theta}) \\ & \propto \tau_n^{\nu_k/2-1} \exp(-\nu_k \tau/2) \tau^{D/2} \det(\boldsymbol{\Sigma}_k)^{-1/2} \exp\left(\frac{1}{2}(\mathbf{x}_n - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_k) \tau_n\right) \\ & = \tau_n^{(\nu_k+D)/2-1} \exp\left(-\frac{1}{2}(\nu_k + (\mathbf{x}_n - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_k)) \tau_n\right) \\ & \propto \text{Ga}(\tau_n; \alpha_{nk}, \beta_{nk}), \end{aligned}$$

where

$$\begin{aligned} \alpha_{nk} &= \frac{\nu_k + D}{2} \\ \beta_{nk} &= \frac{1}{2}(\nu_k + (\mathbf{x}_n - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_k)). \end{aligned}$$

(d) We will first simplify the expected complete log-likelihood,

$$\mathcal{L}(\boldsymbol{\theta}; \boldsymbol{\theta}_{\text{old}}) = \sum_{n=1}^N \mathbb{E}_{p(z_n, \tau_n | \mathbf{x}_n; \boldsymbol{\theta}_{\text{old}})} [\log p(z_n, \tau_n, \mathbf{x}_n; \boldsymbol{\theta})].$$

First recall that,

$$\begin{aligned}
& \log p(z_n, \tau_n, \mathbf{z}_n; \boldsymbol{\theta}) \\
&= \sum_{k=1}^K I[z_n = k] \log p(z_n = k, \tau_n, \mathbf{z}_n; \boldsymbol{\theta}) \\
&= \sum_{k=1}^K I[z_n = k] (\log \pi_k + \log \text{Ga}(\tau_n; \nu_k/2, \nu_k/2) + \log \mathcal{N}(\mathbf{x}_n; \boldsymbol{\mu}_k, \tau_n^{-1} \boldsymbol{\Sigma}_k)) \\
&= \sum_{k=1}^K I[z_n = k] \left(\log \pi_k + \frac{\nu_k}{2} \log(\nu_k/2) - \log \Gamma(\nu_k/2) + (\nu_k/2 - 1) \log(\tau_n) \right. \\
&\quad \left. - \frac{\nu_k}{2} \tau_n - \frac{D}{2} \log(2\pi) - \frac{1}{2} \log \det(\tau_n^{-1} \boldsymbol{\Sigma}_k) - \frac{\tau_n}{2} (\mathbf{x}_n - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_k) \right).
\end{aligned}$$

If we drop terms that don't depend on $\boldsymbol{\theta}$, we have

$$\begin{aligned}
& \log p(z_n, \tau_n, \mathbf{z}_n; \boldsymbol{\theta}) \\
&= \sum_{k=1}^K I[z_n = k] \left(\log \pi_k + \frac{\nu_k}{2} \log(\nu_k/2) - \log \Gamma(\nu_k/2) + (\nu_k/2 - 1) \log(\tau_n) \right. \\
&\quad \left. - \frac{\nu_k}{2} \tau_n - \frac{1}{2} \log \det(\tau_n^{-1} \boldsymbol{\Sigma}_k) - \frac{\tau_n}{2} (\mathbf{x}_n - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_k) \right). \tag{51}
\end{aligned}$$

On the exam, we are only asked to do the M-step for $\{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^K$. We can therefore drop terms that do not depend on $\boldsymbol{\mu}_k$ or $\boldsymbol{\Sigma}_k$. This gives,

$$\begin{aligned}
& \log p(z_n, \tau_n, \mathbf{z}_n; \boldsymbol{\theta}) \\
&= \sum_{k=1}^K I[z_n = k] \left(-\frac{1}{2} \log \det(\tau_n^{-1} \boldsymbol{\Sigma}_k) - \frac{\tau_n}{2} (\mathbf{x}_n - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_k) \right) \\
&= \sum_{k=1}^K I[z_n = k] \left(\frac{D}{2} \log \tau_n - \frac{1}{2} \log \det(\boldsymbol{\Sigma}_k) - \frac{\tau_n}{2} (\mathbf{x}_n - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_k) \right) \\
&= \sum_{k=1}^K I[z_n = k] \left(-\frac{1}{2} \log \det(\boldsymbol{\Sigma}_k) - \frac{\tau_n}{2} (\mathbf{x}_n - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_k) \right),
\end{aligned}$$

where in the last line we dropped the term $\frac{D}{2} \log \tau_n$. We will now take an expectation with respect to $p(z_n, \tau_n | \mathbf{x}_n; \boldsymbol{\theta}_{\text{old}})$. By parts (b) and (c) we have,

$$\mathbb{E}_{p(z_n, \tau_n | \mathbf{x}_n; \boldsymbol{\theta}_{\text{old}})} [I[Z_n = k]] = \omega_{nk}^{\text{old}},$$

and

$$\begin{aligned}
\mathbb{E}_{p(z_n, \tau_n | \mathbf{x}_n; \boldsymbol{\theta}_{\text{old}})} [I[Z_n = k] \tau_n] &= \omega_{nk}^{\text{old}} \mathbb{E}_{\text{Ga}(z_n; \alpha_{nk}^{\text{old}}, \beta_{nk}^{\text{old}})} [\tau_n] \\
&= \frac{\omega_{nk}^{\text{old}} \alpha_{nk}^{\text{old}}}{\beta_{nk}^{\text{old}}}.
\end{aligned}$$

Thus, up to a constant independent of $\{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^K$ we have

$$\begin{aligned} & \mathbb{E}_{p(z_n, \tau_n | \mathbf{x}_n; \boldsymbol{\theta}_{\text{old}})} [\log p(z_n, \tau_n, \mathbf{z}_n; \boldsymbol{\theta})] \\ &= \sum_{k=1}^K \mathbb{E}_{p(z_n, \tau_n | \mathbf{x}_n; \boldsymbol{\theta}_{\text{old}})} \left[I[z_n = k] \left(-\frac{1}{2} \log \det(\boldsymbol{\Sigma}_k) - \frac{\tau_n}{2} (\mathbf{x}_n - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_k) \right) \right] \\ &= \sum_{k=1}^K \omega_{nk}^{\text{old}} \left(-\frac{1}{2} \log \det(\boldsymbol{\Sigma}_k) - \frac{\alpha_{nk}^{\text{old}}}{2\beta_{nk}^{\text{old}}} (\mathbf{x}_n - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_k) \right) \end{aligned}$$

And so, up to terms independent of $\{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^K$, we have

$$\begin{aligned} & \mathcal{L}(\boldsymbol{\theta}; \boldsymbol{\theta}_{\text{old}}) \\ &= \sum_{n=1}^N \sum_{k=1}^K \omega_{nk}^{\text{old}} \left(-\frac{1}{2} \log \det(\boldsymbol{\Sigma}_k) - \frac{\alpha_{nk}^{\text{old}}}{2\beta_{nk}^{\text{old}}} (\mathbf{x}_n - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_k) \right) \\ &= \sum_{k=1}^K \sum_{n=1}^N \omega_{nk}^{\text{old}} \left(-\frac{1}{2} \log \det(\boldsymbol{\Sigma}_k) - \frac{\alpha_{nk}^{\text{old}}}{2\beta_{nk}^{\text{old}}} (\mathbf{x}_n - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_k) \right). \end{aligned}$$

We see that the M-step for $\{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^K$ splits over k . And so,

$$\boldsymbol{\mu}_k^*, \boldsymbol{\Sigma}_k^* \leftarrow \underset{\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k}{\operatorname{argmax}} \sum_{n=1}^N \omega_{nk}^{\text{old}} \left(-\frac{1}{2} \log \det(\boldsymbol{\Sigma}_k) - \frac{\alpha_{nk}^{\text{old}}}{2\beta_{nk}^{\text{old}}} (\mathbf{x}_n - \boldsymbol{\mu}_k)^\top \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_k) \right).$$

If we differentiate with respect to $\boldsymbol{\mu}_k$, we see that $\boldsymbol{\mu}_k^*$ solves,

$$\sum_{n=1}^N \frac{\omega_{nk}^{\text{old}} \alpha_{nk}^{\text{old}}}{\beta_{nk}^{\text{old}}} \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_n - \boldsymbol{\mu}_k^*) = 0 \in \mathbb{R}^D.$$

And so,

$$\boldsymbol{\mu}_k^* \left(\sum_{n=1}^N \frac{\omega_{nk}^{\text{old}} \alpha_{nk}^{\text{old}}}{\beta_{nk}^{\text{old}}} \right) = \sum_{n=1}^N \frac{\omega_{nk}^{\text{old}} \alpha_{nk}^{\text{old}}}{\beta_{nk}^{\text{old}}} \mathbf{x}_n.$$

If we define

$$\gamma_{nk} = \frac{\omega_{nk}^{\text{old}} \alpha_{nk}^{\text{old}} / \beta_{nk}^{\text{old}}}{\sum_{m=1}^N \omega_{mk}^{\text{old}} \alpha_{mk}^{\text{old}} / \beta_{mk}^{\text{old}}},$$

then

$$\boldsymbol{\mu}^* = \sum_{n=1}^N \gamma_{nk} \mathbf{x}_n.$$

Likewise, plugging in $\boldsymbol{\mu}^*$ and differentiating with respect to $\boldsymbol{\Sigma}_k^{-1}$ gives the first order condition,

$$\frac{1}{2} \sum_{n=1}^N \omega_{nk}^{\text{old}} \boldsymbol{\Sigma}_k^* - \frac{\omega_{nk}^{\text{old}} \alpha_{nk}^{\text{old}}}{\beta_{nk}^{\text{old}}} (\mathbf{x}_n - \boldsymbol{\mu}_k^*) (\mathbf{x}_n - \boldsymbol{\mu}_k^*)^\top = 0 \in \mathbb{R}^{D \times D}.$$

And so

$$\boldsymbol{\Sigma}_k^* = \frac{1}{\sum_{n=1}^N \omega_{nk}^{\text{old}}} \sum_{n=1}^N \frac{\omega_{nk}^{\text{old}} \alpha_{nk}^{\text{old}}}{\beta_{nk}^{\text{old}}} (\mathbf{x}_n - \boldsymbol{\mu}_k^*)(\mathbf{x}_n - \boldsymbol{\mu}_k^*)^\top.$$

Optional: In terms of exam requirements, we are now done. We have calculated the M-step for $\{(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\}_{k=1}^K$ which we did by isolating the part of $\mathcal{L}(\boldsymbol{\theta}; \boldsymbol{\theta}^{\text{old}})$ that depended on these parameters. We can do a similar thing for $\boldsymbol{\pi}$. Using (51) we can compute $\log p(z_n, \tau_n, \mathbf{x}_n; \boldsymbol{\theta})$ as a function of only $\boldsymbol{\pi}$ specifically,

$$\log p(z_n, \tau_n, \mathbf{x}_n; \boldsymbol{\theta}) = C + \sum_{k=1}^K I[Z_n = k] \log(\pi_k).$$

Thus, as a function of $\boldsymbol{\pi}$,

$$\begin{aligned} \mathcal{L}(\boldsymbol{\theta}; \boldsymbol{\theta}^{\text{old}}) &= \sum_{n=1}^N \mathbb{E}_{p(z_n, \tau_n | \mathbf{x}_n; \boldsymbol{\theta}^{\text{old}})} \left[\sum_{k=1}^K I[Z_n = k] \log(\pi_k) \right] \\ &= \sum_{n=1}^N \sum_{k=1}^K \omega_{nk}^{\text{old}} \log(\pi_k). \end{aligned}$$

Thus,

$$\pi_k^* = \frac{\sum_{n=1}^N \omega_{nk}}{\sum_{j=1}^K \sum_{n=1}^N \omega_{nj}}.$$

We can be daring and attempt the ν_k updates as well. Up to a constant that does not depend on ν_k we have

$$\begin{aligned} p(z_n, \tau_n, \mathbf{x}_n; \boldsymbol{\theta}) &= C + \sum_{k=1}^K I[z_n = k] \left(\frac{\nu_k}{2} \log(\nu_k/2) - \log \Gamma(\nu_k/2) + \nu_k/2 \log(\tau_n) - \frac{\nu_k}{2} \tau_n \right). \end{aligned}$$

Let $v_k = \nu_k/2$. Thus, as a function of v_k we have

$$\begin{aligned} p(z_n, \tau_n, \mathbf{x}_n; \boldsymbol{\theta}) &= C + \sum_{k=1}^K I[z_n = k] (v_k \log(v_k) - \log \Gamma(v_k) + v_k \log(\tau_n) - v_k \tau_n). \end{aligned}$$

We also have

$$\begin{aligned} \mathbb{E}_{p(z_n, \tau_n | \mathbf{x}_n; \boldsymbol{\theta}^{\text{old}})} [I[Z_n = k]] &= \omega_{nk}^{\text{old}}, \\ \mathbb{E}_{p(z_n, \tau_n | \mathbf{x}_n; \boldsymbol{\theta}^{\text{old}})} [I[Z_n = k] \tau_n] &= \omega_{nk}^{\text{old}} \alpha_{nk}^{\text{old}} / \beta_{nk}^{\text{old}}, \\ \mathbb{E}_{p(z_n, \tau_n | \mathbf{x}_n; \boldsymbol{\theta}^{\text{old}})} [I[Z_n = k] \log(\tau_n)] &= \omega_{nk}^{\text{old}} \mathbb{E}_{\text{Ga}(\tau_n; \alpha_{nk}^{\text{old}}, \beta_{nk}^{\text{old}})} [\log(\tau_n)] \\ &= \omega_{nk}^{\text{old}} (\psi(\alpha_{nk}^{\text{old}}) - \log(\beta_{nk}^{\text{old}})), \end{aligned}$$

where

$$\psi(y) = \frac{d}{dy} \log \Gamma(y),$$

is the digamma function. Thus, as a function of $\mathbf{v} = (v_k)_{k=1}^K$, we have

$$\begin{aligned} \mathcal{L}(\boldsymbol{\theta}; \boldsymbol{\theta}_{\text{old}}) \\ = \sum_{n=1}^N \sum_{k=1}^K \omega_{nk}^{\text{old}} (v_k \log(v_k) - \log \Gamma(v_k) + v_k (\psi(\alpha_{nk}^{\text{old}}) - \log(\beta_{nk}^{\text{old}})) - v_k \alpha_{nk}^{\text{old}} / \beta_{nk}^{\text{old}}). \end{aligned}$$

And so v_k^* maximizes

$$(v_k \log(v_k) - \log \Gamma(v_k)) \sum_{n=1}^N \omega_{nk} + v_k \sum_{n=1}^N \omega_{nk}^{\text{old}} (\psi(\alpha_{nk}^{\text{old}}) - \log(\beta_{nk}^{\text{old}}) - \alpha_{nk}^{\text{old}} / \beta_{nk}^{\text{old}}).$$

Define,

$$\begin{aligned} A_k &= \sum_{n=1}^N \omega_{nk}, \\ B_k &= \sum_{n=1}^N \omega_{nk}^{\text{old}} (\psi(\alpha_{nk}^{\text{old}}) - \log(\beta_{nk}^{\text{old}}) - \alpha_{nk}^{\text{old}} / \beta_{nk}^{\text{old}}). \end{aligned}$$

We are trying to maximize

$$(v_k \log(v_k) - \log \Gamma(v_k)) A_k + v_k B_k.$$

Which has the first order condition

$$(\log(v_k^*) + 1 - \psi(v_k^*)) A_k + B_k = 0$$

So

$$\log(v_k^*) - \psi(v_k^*) = -\frac{A_k + B_k}{A_k}.$$

A plot in Mathematic shows that $v \mapsto \log(v) - \psi(v)$ is strictly decreasing and so the above equation has a unique solution v_k^* . This means that, with a bit of work, we can implement a full EM algorithm for all the parameters $\boldsymbol{\theta}$. An imperfect implementation is available here https://github.com/Michael-Howes/Applied-Coaching/blob/main/Code/student_mixture.ipynb.

Question 6: Principal component regression with ℓ_1 -penalties

Key ideas:

- The role of the SVD in least squares regression and ridge regression.

- Reducing a multi-parameter optimization problem into several one-dimensional problems.
- ℓ_1 -penalties inducing sparsity.

(a) The function,

$$S_\lambda(t) := \operatorname{argmin}_{s \in \mathbb{R}} \left\{ \frac{1}{2}(s - t)^2 + \lambda|s| \right\},$$

is the *soft-threshold* function. It is given by,

$$S_\lambda(t) = \begin{cases} t + \lambda & \text{if } t < -\lambda, \\ 0 & \text{if } -\lambda \leq t \leq \lambda, \\ t - \lambda & \text{if } t > \lambda. \end{cases}$$

In convex optimization it is called the *proximal function* for the absolute value function $|\cdot|$. On the qualifying exam, you are not asked to derive the above expression for $S_\lambda(t)$, but we will go through a derivation here.

Define $f_t(s) = \frac{1}{2}(s - t)^2 + \lambda|s|$. Note that,

$$f_t(s) = \begin{cases} \frac{1}{2}(s - t)^2 - \lambda s & \text{if } s \leq 0, \\ \frac{1}{2}(s - t)^2 + \lambda s & \text{if } s > 0. \end{cases}$$

The above function is convex as it is the maximum of two convex functions. We'll now consider different cases.

If $t < -\lambda$, then $s^* := t + \lambda < 0$. For $s < 0$, $f_t(s)$ is differentiable with derivative $s - t - \lambda$. Thus, $s^* = t + \lambda < 0$ solves the first order conditions and is a minimizer of $f_t(s)$.

Likewise, if $t > \lambda$, the $s^* := t - \lambda > 0$. For $s > 0$, $f_t(s)$ is differentiable with derivative $s - t + \lambda$. Thus, $s^* = t - \lambda > 0$ is a minimizer of $f_t(s)$.

Finally, if $-\lambda \leq t \leq \lambda$, then the first order conditions have no stationary points and so the minimizer must be where $f_t(s)$ is non-differentiable. That is, $s^* = 0$.

(b) We are asked to find,

$$\hat{\beta}_\lambda := \operatorname{argmin}_{b \in \mathbb{R}^d} \left\{ \frac{1}{2} \|Xb - y\|_2^2 + \sum_{j=1}^d \lambda_j |v_j^\top b| \right\},$$

where $V = [v_1, \dots, v_d] \in \mathbb{R}^{d \times d}$ are the right singular vectors of X . Specifically we are given,

$$X = UTV^\top,$$

where

- $U = [u_1, \dots, u_d] \in \mathbb{R}^{n \times d}$ with $U^\top U = I_d$.
- Γ is a $d \times d$ diagonal matrix of singular values $\Gamma = \text{diag}(\gamma_1, \dots, \gamma_d)$ with $\gamma_1 \geq \dots \geq \gamma_d > 0$.
- $V = [v_1, \dots, v_d] \in \mathbb{R}^{d \times d}$ satisfies $V^\top V = VV^\top = I_d$.

For all $b \in \mathbb{R}^d$, we have

$$\begin{aligned} Xb - Y &= U\Gamma V^\top b - Y \\ &= U\Gamma V^\top b - UU^\top Y - (I - UU^\top)Y \\ &= U(\Gamma V^\top b - U^\top Y) - (I - UU^\top)Y. \end{aligned}$$

Note that

$$U^\top (I - UU^\top) = U^\top - U^\top = 0,$$

and

$$(I - UU^\top)U = U - U = 0.$$

The vectors $U(\Gamma V^\top b - U^\top Y) \in \mathbb{R}^n$ and $(I - UU^\top)Y \in \mathbb{R}^n$ are therefor orthogonal to each other. Thus,

$$\|Xb - Y\|_2^2 = \|U(\Gamma V^\top b - U^\top Y) - (I - UU^\top)Y\|_2^2 = \|U(\Gamma V^\top b - U^\top Y)\|_2^2 + \|(I - UU^\top)Y\|_2^2.$$

Since the second term does not depend on b , we can conclude that $\hat{\beta}_\lambda$ solves

$$\hat{\beta}_\lambda = \underset{b \in \mathbb{R}^n}{\text{argmin}} \left\{ \frac{1}{2} \|U(\Gamma V^\top b - U^\top Y)\|_2^2 + \sum_{j=1}^d \lambda_j |v_j^\top b| \right\}.$$

Since $U^\top U = I_d$, we have that $\|Ux\|_2^2 = \|x\|_2^2$ for all $x \in \mathbb{R}^d$. Thus,

$$\|U(\Gamma V^\top b - U^\top Y)\|_2^2 = \|\Gamma V^\top b - U^\top Y\|_2^2.$$

Let $c = V^\top b \in \mathbb{R}^d$ so that $c_j = v_j^\top b$. Since $b = Vc$, we have that $\hat{\beta}_\lambda$ satisfies,

$$\hat{\beta}_\lambda = Vc^*,$$

$$\text{where } c^* = \underset{c \in \mathbb{R}^d}{\text{argmin}} \left\{ \|\Gamma c - U^\top Y\|_2^2 + \sum_{j=1}^d \lambda_j |c_j| \right\}.$$

Since Γ is diagonal we have that,

$$\begin{aligned} \|\Gamma c - U^\top Y\|_2^2 + \sum_{j=1}^d \lambda_j |c_j| &= \sum_{j=1}^d (\gamma_j c_j - u_j^\top Y)^2 + \sum_{j=1}^d \lambda_j |c_j| \\ &= \sum_{j=1}^d (\gamma_j c_j - u_j^\top Y)^2 + \lambda_j |c_j| \\ &= \sum_{j=1}^d \gamma_j^2 \left[\left(c_j - \frac{u_j^\top Y}{\gamma_j} \right)^2 + \frac{\lambda_j}{\gamma_j^2} |c_j| \right]. \end{aligned}$$

The objective function for c is therefor separable over the coordinates c_j . We can thus calculate c^* by solving p one-dimensional optimization problems. Specifically,

$$\begin{aligned} c_j^* &= \operatorname{argmin}_{c_j} \left\{ \gamma_j^2 \left[\left(c_j - \frac{u_j^\top Y}{\gamma_j} \right)^2 + \frac{\lambda_j}{\gamma_j^2} |c_j| \right] \right\} \\ &= \operatorname{argmin}_{c_j} \left\{ \left(c_j - \frac{u_j^\top Y}{\gamma_j} \right)^2 + \frac{\lambda_j}{\gamma_j^2} |c_j| \right\} \\ &= S_{\frac{\lambda_j}{\gamma_j^2}} \left(\frac{u_j^\top Y}{\gamma_j} \right) \end{aligned}$$

where $S_\lambda(t)$ is the soft-threshold function as in part (a). It follows that,

$$\widehat{\beta}_\lambda = V c^* = \sum_{j=1}^d S_{\frac{\lambda_j}{\gamma_j^2}} \left(\frac{u_j^\top Y}{\gamma_j} \right) v_j$$

- (c) We now assume that $\gamma_j = 1$ for all j . We know that $\widehat{\beta}_{\text{ridge}}$ can also be described in terms of the SVD of X . Specifically, in the special case when $\gamma_j = 1$ for all j

$$\widehat{\beta}_{\text{ridge}} = \sum_{j=1}^d \frac{1}{1 + \lambda_{\text{ridge}}} u_j^\top Y v_j.$$

And, from part (b),

$$\widehat{\beta}_\lambda = \sum_{j=1}^d S_{\lambda_j} (u_j^\top Y) v_j$$

We see that when written in the basis v_1, \dots, v_d , the ridge coefficients and principal-components ℓ_1 -regularized are both shrunk version of the OLS coefficients $a_j = u_j^\top Y$. Specifically,

- Ridge coefficients: $\frac{1}{1 + \lambda_{\text{ridge}}} a_j$.
- ℓ_1 -regularized components: $S_{\lambda_j}(a_j)$.

The performance of these two estimators will depend on the expansion of the true parameter vector β in the basis v_1, \dots, v_d . If when we write $\beta = \sum_{j=1}^d b_j v_j$, many of the b_j 's are zero. Then we will expect the ℓ_1 -regularized estimator to perform better. This is because the soft threshold function $S_{\lambda_j}(a_j)$ will set some coefficients equal to zero. On the other the ridge shrinkage simply scales each a_j by a constant amount.

References

- James H Albert and Siddhartha Chib. Bayesian analysis of binary and polychotomous response data. *Journal of the American statistical Association*, 88(422):669–679, 1993.
- Simon Anders and Wolfgang Huber. Differential expression analysis for sequence count data. *Genome Biology*, 11(10):R106, 2010.
- Luigi Augugliaro, Antonino Abbruzzo, and Veronica Vinciotti. ℓ_1 -penalized censored Gaussian graphical model. *Biostatistics*, 21(2):e1–e16, 2020.
- Stephen Bates, Trevor Hastie, and Robert Tibshirani. Cross-validation: what does it estimate and how well does it do it? *arXiv preprint arXiv:2104.00673*, 2022.
- Javad Behboodian. On the modes of a mixture of two normal distributions. *Technometrics*, 12(1):131–139, 1970.
- A. C. Davison and D. V. Hinkley. *Further Ideas*, page 70–135. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 1997. doi: 10.1017/CBO9780511802843.004.
- Persi Diaconis and Donald Ylvisaker. Quantifying prior opinion. *Bayesian Statistics 2. Proc. 2nd Valencia Int’l Meeting, 9-83. J. M. Bernardo, M. H. Degroot, D. V. Lindley, A. F. M. Smith (eds.) North-Holland, Amsterdam*, 15(12):133–156, 1985.
- Thomas J. DiCiccio and Bradley Efron. Bootstrap confidence intervals. *Statistical Science*, 11(3):189 – 228, 1996. doi: 10.1214/ss/1032280214. URL <https://doi.org/10.1214/ss/1032280214>.
- Bradley Efron. Tweedie’s formula and selection bias. *Journal of the American Statistical Association*, 106(496):1602–1614, 2011.
- Bradley Efron. Empirical Bayes deconvolution estimates. *Biometrika*, 103(1):1–20, 2016.
- Bradley Efron and Trevor Hastie. *Computer age statistical inference*, volume 5. Cambridge University Press, 2016.
- Chen Gao, Yunzhang Zhu, Xiaotong Shen, and Wei Pan. Estimation of multiple networks in gaussian mixture models. *Electronic journal of statistics*, 10:1133, 2016.
- Botao Hao, Will Wei Sun, Yufeng Liu, and Guang Cheng. Simultaneous clustering and estimation of heterogeneous graphical models. *The Journal of Machine Learning Research*, 18(1):7981–8038, 2017.

- Trevor Hastie and Robert Tibshirani. Efficient quadratic regularization for expression arrays. *Biostatistics*, 5(3):329–340, 2004.
- Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The elements of statistical learning: data mining, inference, and prediction*. Springer Science & Business Media, 2009.
- Jason D Lee and Jonathan E Taylor. Exact post model selection inference for marginal screening. In *Advances in neural information processing systems*, pages 136–144, 2014.
- Jason D Lee, Dennis L Sun, Yuekai Sun, Jonathan E Taylor, et al. Exact post-selection inference, with application to the lasso. *The Annals of Statistics*, 44(3):907–927, 2016.
- Thomas A Louis. Finding the observed information matrix when using the EM algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 44(2):226–233, 1982.
- Karim Lounici. High-dimensional covariance matrix estimation with missing observations. *Bernoulli*, 20(3):1029–1058, 2014.
- K V Mardia, J M Bibby, and J T Kent. *Multivariate analysis*. 1979. ISBN 0124712509 0124712525 0124712525. URL <http://www.loc.gov/catdir/toc/els031/79040922.html>.
- R. G. Miller, B. Efron, B. W. Brown, and L. E. Moses. *Biostatistics Case Book, Volume One*. Wiley, 1976. URL <https://statistics.stanford.edu/technical-reports/biostatistics-case-book-volume-one>.
- Joris M Mooij, Jonas Peters, Dominik Janzing, Jakob Zscheischler, and Bernhard Schölkopf. Distinguishing cause from effect using observational data: methods and benchmarks. *The Journal of Machine Learning Research*, 17(1):1103–1204, 2016.
- Jean Morrison and Noah Simon. Rank conditional coverage and confidence intervals in high-dimensional problems. *Journal of Computational and Graphical Statistics*, 27(3):648–656, 2018.
- Guillaume Obozinski, Martin J. Wainwright, and Michael I. Jordan. Support union recovery in high-dimensional multivariate regression. *The Annals of Statistics*, 39(1):1–47, 2011. doi: 10.1214/09-AOS776. URL <https://doi.org/10.1214/09-AOS776>.
- Nicholas G Polson, James G Scott, and Jesse Windle. Bayesian inference for logistic models using Pólya–Gamma latent variables. *Journal of the American statistical Association*, 108(504):1339–1349, 2013.

- Peter E Rossi, Greg M Allenby, and Rob McCulloch. *Bayesian statistics and marketing*. John Wiley & Sons, 2005.
- Yiyuan She and Art B Owen. Outlier detection using nonconvex penalized regression. *Journal of the American Statistical Association*, 106(494):626–639, 2011.
- Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.
- Georges M Tallis. The moment generating function of the truncated multi-normal distribution. *Journal of the Royal Statistical Society: Series B (Methodological)*, 23(1):223–229, 1961.
- Stefan Wager, Sida Wang, and Percy S Liang. Dropout training as adaptive regularization. In *Advances in neural information processing systems*, pages 351–359, 2013.
- Asaf Weinstein, William Fithian, and Yoav Benjamini. Selection adjusted confidence intervals with more power to determine the sign. *Journal of the American Statistical Association*, 108(501):165–176, 2013.
- Daniela M Witten, Robert Tibshirani, and Trevor Hastie. A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis. *Biostatistics*, 10(3):515–534, 2009.
- Qingyuan Zhao, Dylan S Small, and Weijie Su. Multiple testing when many p-values are uniformly conservative, with application to testing qualitative interaction in educational interventions. *Journal of the American Statistical Association*, 114(527):1291–1304, 2019.