STAT 154 Spring 2019: Sample Mid-term Exam

Instructor: Prof. Bin Yu
Mar 21, 8:10 AM–9:30 AM

Your Name: _____ Student ID: _____

Maximum Points: 50, Time: 80 minutes

Instructions

- Do not turn the page until you are told do so.
- **WRITE your student ID CLEARLY on each page** in the space provided at the top.
- This exam has **5 questions** with **24 parts** in total.
- Your answer **will be graded only if** it is written in the space provided after the question. You may use the blank pages at the end for rough work, your rough work will **NOT** be graded.
- Try not to get stuck in a particular problem.
- *When time is up*, please take your exam in your **left hand** and raise it. When asked pass it on to your left (facing the board). Please maintain the sequence of your seating. Instructor/GSI will collect the copies from the leftmost student in the row.
- As promised, there is a help-sheet (last page) for your comfort.

Pre-Exam Questions

1. What is (are) your favorite movie(s)?

2. What do you like the most about this class?

1 True or False (with justification, 10 pts)

Please justify your reasoning in one line.

(a) For a model to make meaningful predictions on the future data, we need some similarity between the representative data that was used to build the model and the future data.

True (HW1) - The new data has to be similar
Else → wrong conclusion.

(b) If the rank of a $d \times d$ projection matrix is k, then its trace is equal to $d - k$.

False. (tr = k)

(c) K-means is guaranteed to return the global minimizer of the following objective:

$$\min_{C_1,\ldots,C_K} \min_{\mu_1,\ldots,\mu_K} \sum_{k=1}^{K} \sum_{x_i \in C_k} \|x_i - \mu_k\|_2^2.$$

~~True~~
False (local minima) ← Non-convex problem

(d) For any $d \times d$ matrix, we have $\nabla_\theta(\theta^\top \mathbf{A}\theta) = 2\mathbf{A}\theta$.

False

$(A + A^T)\theta$.

$\nabla^2_\theta \theta^T A \theta = (A + A^T)$

(e) If some features ($\{x_1, \ldots, x_n\}$) are linearly dependent, at least one singular value of the feature matrix $\mathbf{X}$ is zero.

True (X is not full rank if cols are linearly dependent).

(f) Lasso always produces features that are subset of original features.

True Some coefficients → zero.

(g) Number of features in a model or the number of iterations when using a method like gradient descent can act as regularization for training the model.

True Complexity of model is affected.

(h) Gradient descent on the ordinary least squares objective function is guaranteed to converge to the ridge estimate as long as the step size is positive.

False Converge to OLS estimate. Stepsize cannot be too large.

(i) The output of PCA is a new representation of the data that is always of terms of the principal components that are best at predicting the output variable.

False, PCA is unsupervised
↳ No for prediction.
No explicit output y.

(j) When the features in the data matrix have same units and are comparable to each other, simply centering them and not scaling them before doing PCA is a good idea.

~~False~~. True - Only scale when features have different units → so that data is comparable.

2 EM with Poisson mixture (10 pts)

We work with the following simple two mixture model:

$$\begin{aligned} Z &\sim \text{Bernoulli}(1-w)+1 \\ X|Z=1 &\sim \text{Poisson}(\lambda) \quad \text{and} \\ X|Z=2 &\sim \text{Poisson}(\mu) \end{aligned} \tag{1}$$

where Z denotes the label of the Poisson distribution from which X is drawn. Given n observations $\{x_1, \ldots, x_n\}$ only for X (i.e., the labels are unobserved), our goal is to infer the maximum-likelihood parameters for λ_1, λ_2, and w using EM.

Note that for a random variable $Y \sim \text{Poisson}(\lambda)$, we have

$$\mathbb{P}(Y=k) = e^{-\lambda}\frac{\lambda^k}{k!}, \quad \text{for } k = 0, 1, \ldots.$$

(a) (2 pts) **Derive the complete log-likelihood for the observed data , i.e., derive the expression for $\sum_{i=1}^{n} \log \mathbb{P}(X = x_i; \lambda, \mu, w)$ in terms of the parameters and the data $\{x_1, \ldots, x_n\}$.**

(b) (2 pts) **Write down the expression for the lower bound $\mathcal{F}(\lambda, \mu, w, q)$ where $q = (q_1, \ldots, q_n)$ where $q = (q_1, \ldots, q_n)$ and $q_i = p(Z_i = 1)$ denotes the parameter for an appropriate distribution on the hidden label Z_i.**

(c) (2 pts) Given the iterates w_t, λ_t and μ_t, compute the E-step updates, i.e., **write the expressions for**

$$q^{t+1} = \arg\max_q \mathcal{F}(\lambda_t, \mu_t, w_t, q)$$

No need to show any derivation. Simply writing the expressions suffices. *Hint: q_i^{t+1} is related to a posterior distribution.*

(d) (4 pts) **Derive the M-step updates, i.e., compute the expressions**

$$\lambda_{t+1}, \mu_{t+1}, w_{t+1} = \arg\max_{\lambda,\mu,w} \mathcal{F}(\lambda, \mu, w, q^{t+1}),$$

Hint: Some gradients have to be set to zero.

3 Weighted Least Squares (8 pts)

The weighted least squares (WLS) is a slight modification on least squares:

$$\min_{\theta} \sum_{i=1}^{n} w_i (x_i^\top \theta - y_i)^2 \tag{2}$$

i.e., the i-th term in the least squares objective is weighted by a term w_i.

Define the matrices:

$$\mathbf{W} = \begin{bmatrix} w_1 & & & \\ & w_2 & & \\ & & \ddots & \\ & & & w_n \end{bmatrix} \in \mathbb{R}^{n \times n}, \quad \mathbf{X} = \begin{bmatrix} \mathbf{x}_1^\top \\ \mathbf{x}_2^\top \\ \vdots \\ \mathbf{x}_n \end{bmatrix} \in \mathbb{R}^{n \times d} \quad \text{and} \quad \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \in \mathbb{R}^n$$

(a) (2 pts) **Write the objective in equation** (2) **using the matrix-vector notation.** Your answer should be in terms of $\mathbf{W}, \mathbf{X}, \mathbf{y}$ and θ.

(b) (3 pts) Let the matrix $\sqrt{\mathbf{W}}\mathbf{X}$ be full column rank. **Derive the closed form solution for the WLS estimate obtained by minimizing the objective in equation** (2)**.** *Hint: You may use the OLS formula for this part.*

(c) (3 pts) Consider the following linear model:

$$y_i = x_i^\top \theta + \varepsilon_i, \tag{3}$$

where $\varepsilon_i \sim \mathcal{N}(0, \sigma_i^2)$. Assume that the noise variables ε_i are independent (not identically distributed) of each other and independent of x_i's. Also assume that the variances σ_i^2 are known. Given n independent samples $\{(x_i, y_i), i = 1, \dots n\}$ from the model (3), **show that computing the maximum likelihood estimate of θ for this data is equivalent to solving a weighted least squares problem. What are the weights w_i?**

4 Ridge Regression vs PCA (12 pts)

Assume we are given n training data points $(\mathbf{x}_i, y_i)$. We collect the responses into $\mathbf{y} \in \mathbb{R}^n$, and the features into the matrix $\mathbf{X} \in \mathbb{R}^{n \times d}$ where the rows are the $d-$dimensional feature vectors $\mathbf{x}_i^\top$ corresponding to each training point. Furthermore, assume that $\frac{1}{n}\sum_{i=1}^n \mathbf{x_i} = \mathbf{0}$, $n > d$ and $\mathbf{X}$ has rank d.

In this problem we want to compare two procedures: The first is ridge regression with hyper-parameter λ, while the second is applying ordinary least squares after using PCA to reduce the feature dimension from d to k (we give this latter approach the short-hand name k-PCA-OLS where k is the hyper-parameter).

Notation: The singular value decomposition of $\mathbf{X}$ reads $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$ where $\mathbf{U} \in \mathbb{R}^{n \times n}$, $\mathbf{\Sigma} \in \mathbb{R}^{n \times d}$ and $\mathbf{V} \in \mathbb{R}^{d \times d}$. We denote by $\mathbf{u}_j$ the n-dimensional column vectors of $\mathbf{U}$ and by $\mathbf{v}_j$ the $d-$dimensional column vectors of $\mathbf{V}$. Furthermore the diagonal entries $\sigma_j = \Sigma_{j,j}$ of $\mathbf{\Sigma}$ satisfy $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_d > 0$. For notational convenience, assume that $\sigma_j = 0$ for $j > d$.

(a) (4 pts) It turns out that the ridge regression optimizer (with $\lambda > 0$) in the $\mathbf{V}$-transformed coordinates

$$\widehat{\theta}_{\text{ridge}} = \arg\min_{\theta} \|\mathbf{X}\mathbf{V}\theta - \mathbf{y}\|_2^2 + \lambda\|\theta\|_2^2$$

has the following expression:

$$\widehat{\theta}_{\text{ridge}} = \text{diag}(\frac{\sigma_j}{\lambda + \sigma_j^2})\mathbf{U}^\top\mathbf{y}. \tag{4}$$

Use $\widehat{y}_{test} = \mathbf{x}_{test}^\top \mathbf{V}\widehat{\theta}_{\text{ridge}}$ to denote the resulting prediction for a hypothetical $\mathbf{x}_{test}$. Using (4) and the appropriate scalar $\{\alpha_i\}$, this can be written as:

$$\widehat{y}_{test} = \mathbf{x}_{test}^\top \sum_{j=1}^{d} \mathbf{v}_j \alpha_j \mathbf{u}_j^\top \mathbf{y}. \tag{5}$$

What are the $\alpha_j \in \mathbb{R}$ for this to correspond to (4) from ridge regression? Show your work.

(b) (5 pts) Suppose that we do k-PCA-OLS — i.e. ordinary least squares on the reduced k-dimensional feature space obtained by projecting the raw feature vectors onto the $k < d$ principal components of the covariance matrix $\mathbf{X}^\top \mathbf{X}$. Use $\widehat{y}_{test}$ to denote the resulting prediction for a hypothetical $\mathbf{x}_{test}$,

It turns out that the learned k-PCA-OLS predictor can be written as:

$$\widehat{y}_{test} = \mathbf{x}_{test}^\top \sum_{j=1}^{d} \mathbf{v}_j \alpha_j \mathbf{u}_j^\top \mathbf{y}. \tag{6}$$

Give the $\alpha_j \in \mathbb{R}$ coefficients for k-PCA-OLS. Show work. These α_j's would be different compared to the previous part.

Hint 1: some of these α_j will be zero. Also, if you want to use the compact form of the SVD, feel free to do so if that speeds up your derivation.

Hint 2: some inspiration may be possible by looking at the next part for an implicit clue as to what the answer might be.

(c) (3 pts) For the following part, $d = 5$. The following $\alpha := (\alpha_1, \ldots, \alpha_5)$ (written out to two significant figures) are the results of OLS (i.e. what we would get from ridge regression in the limit $\lambda \to 0$), λ-ridge-regression, and k-PCA-OLS for some $\mathbf{X}, \mathbf{y}$ (identical for each method) and $\lambda = 1, k = 3$. **Write down which procedure was used for each of the three sub-parts below.**

We hope this helps you intuitively see the connection between these three methods.

Hint: It is not necessary to find the singular values of $\mathbf{X}$ *explicitly, or to do any numerical computations at all.*

(i) $\alpha = (0.01, 0.1, 0.5, 0.1, 0.01)$

(ii) $\alpha = (0.01, 0.1, 1, 0, 0)$

(iii) $\alpha = (0.01, 0.1, 1, 10, 100)$

ii) PCA - Exact zeros

i) Ridge - Should have smaller coefficients.

iii) OLS -

5 EDA with Ames Data (10 pts)

In the two figures below, you are given two scatter-plots with the same data namely, two features "SalePrice" and "Gr.Liv.Area" of the Ames dataset.

(a) (2 pts) Figure 1 was generated using ggplot2 library with the command

```
ggplot(x) + geom_point(aes(x=Ames[, ``Gr.Liv.Area''], y=Ames , ``SalePrice''))
```

but without changing any parameters. However to produce Figure 2, certain aesthetic changes were done (no data processing was done). Can you guess what kind of changes in the aesthetics were made? Which figure is more informative? And why?

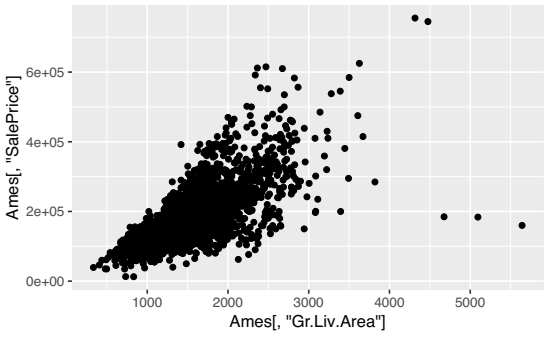


Fig 1.

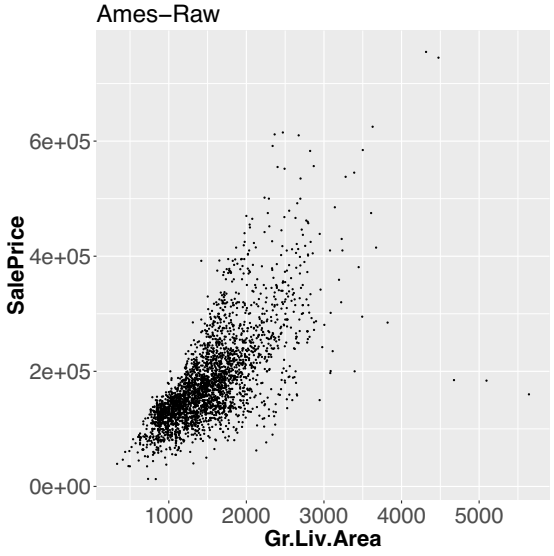


Fig. 2

- Point Size
- X Y- Axis Ratio
- Title
- Font Size.

(b) (2 pts) The data from the previous part was *transformed* in **two different ways** to obtain the following two figures. **Can you guess what kind of transformations were made to obtain these plots?** To answer this question, you should compare these figures with Figure 2 in the previous part.

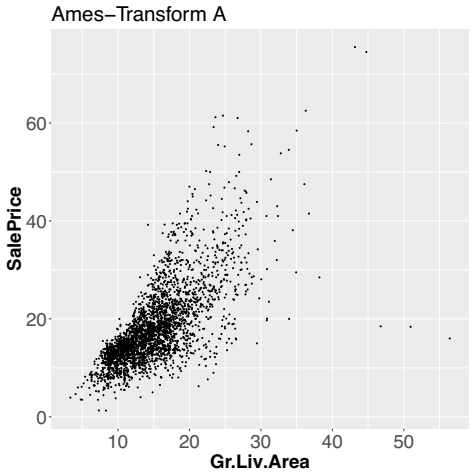


Fig 3.

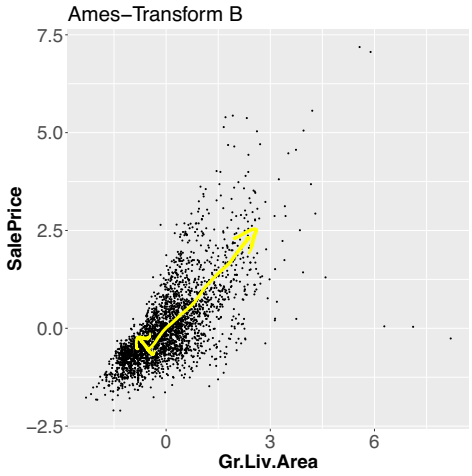


Fig. 4

- Scaling

- Centering
- Scaling

} Standardizing.

(c) (3 pts) **Between the datasets Ames-Raw(Figure 2), Ames-Transform A (Figure 3) and Ames-Transform B (Figure 3), which version of the dataset is most suitable for doing a PCA analysis? Why? Also plot the principal vectors on the figure that you think is most appropriate.**

Figure 4

- X Y Axes are of same unit
- Centered & Scaling

(d) (3 pts) **Do you think using K-means or EM would provide some insight with any of the datasets (Figure 2/3/4)? Justify.**

No. No visible clusters.

Help-sheet

- Given features $\mathbf{X} \in \mathbb{R}^{n \times d}$ and responses $\mathbf{y} \in \mathbb{R}^n$, if the feature matrix $\mathbf{X}$ is full column rank, the OLS solution is given by

$$\widehat{\beta}^{\text{OLS}} = \arg\min_{\beta} \|\mathbf{X}\beta - \mathbf{y}\|_2^2 = (\mathbf{X}^\top \mathbf{X})^{-1}\mathbf{X}^\top \mathbf{y}.$$

- The pdf/PMF of a few distributions is given below:

Distribution	**Notation**	**pdf** (p) **/ PMF** $(\mathbb{P})$
Multi-variate Gaussian	$Z \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$	$p(Z = \mathbf{z}) = \frac{1}{\sqrt{\det(2\pi\boldsymbol{\Sigma})}} \exp\left(-\frac{(\mathbf{z}-\boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1}(\mathbf{z}-\boldsymbol{\mu})}{2}\right)$
Exponential	$Z \sim \text{Exponential}(\lambda)$	$p(Z = z) = \lambda e^{-\lambda z}, \quad z \geq 0.$
Bernoulli	$Z \sim \text{Bernoulli}(\alpha)$	$\mathbb{P}(Z = z) = \alpha^z (1-\alpha)^{(1-z)}, \quad z \in \{0, 1\}$
Poisson	$Z \sim \text{Poisson}(\lambda)$	$\mathbb{P}(Z = z) = e^{-\lambda} \frac{\lambda^z}{z!}, \quad z \in \{0, 1, 2, \ldots\}.$

Fun Space

Feel free to draw/write something if you want to or give us suggestions or complaints. You can also use this space to report anything suspicious that you might have noticed.

Space for rough work (will not be graded)

Space for rough work (will not be graded)

2 EM with Poisson mixture (10 pts)

We work with the following simple two mixture model:

$$\begin{aligned} Z &\sim \text{Bernoulli}(1-w)+1 \\ X|Z=1 &\sim \text{Poisson}(\lambda) \quad \text{and} \\ X|Z=2 &\sim \text{Poisson}(\mu) \end{aligned} \tag{1}$$

where Z denotes the label of the Poisson distribution from which X is drawn. Given n observations $\{x_1, \ldots, x_n\}$ only for X (i.e., the labels are unobserved), our goal is to infer the maximum-likelihood parameters for λ_1, λ_2, and w using EM.

Note that for a random variable $Y \sim \text{Poisson}(\lambda)$, we have

$$\mathbb{P}(Y=k) = e^{-\lambda}\frac{\lambda^k}{k!}, \quad \text{for } k = 0, 1, \ldots.$$

(a) (2 pts) **Derive the complete log-likelihood for the observed data , i.e., derive the expression for $\sum_{i=1}^{n} \log \mathbb{P}(X = x_i; \lambda, \mu, w)$ in terms of the parameters and the data $\{x_1, \ldots, x_n\}$.**

$A^{-1} \longrightarrow A^T A$ is PD

$Ax \neq 0$

$A \neq 0$

$x^T A^T A x$

$= (Ax)^T Ax > 0$

$\sum (A_{ii} x_i)^2 > 0$

$\begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} \begin{bmatrix} \quad \end{bmatrix}$

$x^T A^T A x$

$Ax = n \times 1 = (Ax)^T Ax \leftarrow \|Ax\|_2^2 \geq 0$

$\|Ax\|_2^2 > 0$

$\sum (a_i x_i)^2$

$\underset{n \times n}{A} \quad \underset{n \times 1}{x}$

$\begin{bmatrix} A \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$

$Ax^T = [a_1 \; a \; \cdots] \begin{bmatrix} a_1 \\ \vdots \\ a_n \end{bmatrix}$

$Ax = Ax$: $\nabla_x x^T a = a^T$

$\arg\min \|X\theta - y\|_2^2$ — $a^T x = a^T$

$\mathcal{L}(\theta) \geq F(\theta, q)$ $\rightarrow \mathcal{L}(\theta) = F(\theta, q)$

E-Step: $q^{t+1} \in \operatorname{argmax} F(\theta^t, q)$

M-Step: $\theta^{t+1} \in \operatorname{argmax} F(\theta, q^{t+1})$

$\downarrow$

q^{t+1} is a maximizer of $F(\theta^t, q) = \mathcal{L}(\theta^t)$

E-Step: $q_i^{t+1}(k) = p(z_i = k \mid X = x_i)$

$$\sum_{i=1}^{n} \log p(X = x_i; \theta) = \sum_{i=1}^{n} \sum_{k=1}^{2} P(z_i = k \mid X = x_i; \theta^t) \log \frac{P(X = x_i, z_i = k; \theta^t)}{P(z_i = k \mid X = x_i; \theta^t)}$$

$k=1$:

$$q_i^{t+1}(k) = P(z = 1 \mid X = x_i; \theta^t) = \frac{P(X = x_i, z = 1; \theta^t)}{P(X = x_i; \theta^t)}$$

$$= \frac{w_1 e^{-\lambda_1} \frac{\lambda_1^{x_i}}{(x_i)!}}{w_k e^{-\lambda_k} \frac{\lambda_k^{x_i}}{(x_i)!} + (1-w) e^{-\mu} \frac{\mu^{x_i}}{(x_i)!}}$$

$k=2$:

$$q_i^{t+1}(2) = 1 - q_i^{t+1}(1)$$

$$F(\theta, q^{(t+1)}) := \sum_{i=1}^{n} q_i^{t+1}(1) \log\left[w - \lambda + x_i \log\lambda + q_i^{t+1}(2) \underbrace{\log\left((1-w) e^{-\mu} \frac{\mu^{x_i}}{(x_i)!}\right)}_{\log(1-w) - \mu + x_i \log\mu.}\right] + C$$

$$\nabla_w F = 0 \Rightarrow \sum_{i=1}^{n} \left(\frac{q_i^{t+1}(1)}{w} - \frac{q_i^{t+1}(2)}{1-w} = 0 \right)$$

$$\Rightarrow \sum_{i=1}^{n} q_i^{t+1}(i) - w\left(\sum_{i=1}^{n} q_i^{t+1}(1) + \sum_{i=1}^{n} q_i^{t+1}(2) \right) = 0$$

$$\Rightarrow w_{t+1} = \frac{\sum_{i=1}^{n} q_i^{t+1}(1)}{n}$$

$$\nabla_\lambda F = 0 \Rightarrow \sum_{i=1}^{n} q_i^{t+1}(1)\left(-1 + \frac{x_i}{\lambda}\right) = 0$$

$$\Rightarrow \sum_{i=1}^{n} \lambda q_i^{(t+1)}(1) = \sum_{i=1}^{n} x_i q_i^{(t+1)}(1)$$

$$\Rightarrow \lambda_{t+1} = \frac{\sum_{i=1}^{n} x_i q_i^{(t+1)}(1)}{\sum_{i=1}^{n} q_i^{t+1}(1)}$$

$$\mu_{t+1} = \frac{\sum_{i=1}^{n} x_i q_i^{(t+1)}(2)}{\sum_{i=1}^{n} q_i^{(t+1)}(2)}$$

(b) (2 pts) **Write down the expression for the lower bound $\mathcal{F}(\lambda, \mu, w, q)$ where $q = (q_1, \ldots, q_n)$ where $q = (q_1, \ldots, q_n)$ and $q_i = p(Z_i = 1)$ denotes the parameter for an appropriate distribution on the hidden label Z_i.**

3 Weighted Least Squares (8 pts)

The weighted least squares (WLS) is a slight modification on least squares:

$$\min_{\theta} \sum_{i=1}^{n} w_i (x_i^\top \theta - y_i)^2 \tag{2}$$

i.e., the i-th term in the least squares objective is weighted by a term w_i.

Define the matrices:

$$\mathbf{W} = \begin{bmatrix} w_1 & & & \\ & w_2 & & \\ & & \ddots & \\ & & & w_n \end{bmatrix} \in \mathbb{R}^{n \times n}, \quad \mathbf{X} = \begin{bmatrix} \mathbf{x}_1^\top \\ \mathbf{x}_2^\top \\ \vdots \\ \mathbf{x}_n \end{bmatrix} \in \mathbb{R}^{n \times d} \quad \text{and} \quad \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \in \mathbb{R}^n$$

(a) (2 pts) **Write the objective in equation** (2) **using the matrix-vector notation.** Your answer should be in terms of $\mathbf{W}, \mathbf{X}, \mathbf{y}$ and θ.

$\sum_{i=1}^{n} w_i (x_i^\top \theta - y_i)^2$

$= \sum_{i=1}^{n} (\underbrace{\sqrt{w_i}\, x_i^\top}_{\tilde{x}_i} \theta - \underbrace{\sqrt{w_i}\, y_i}_{\tilde{y}_i})^2$

$= \sum_{i=1}^{n} (\tilde{x}_i^\top \theta - \tilde{y}_i)^2$

$= \| \tilde{X}\theta - \tilde{y} \|_2^2$

$= (\tilde{X}\theta - \tilde{y})^\top \begin{pmatrix} \tilde{X}\theta \\ \tilde{y} \end{pmatrix}$

$= (X\theta - y)^\top \sqrt{W}\sqrt{W}(X\theta - y)$

$= (X\theta - y)^\top W (X\theta - y)$

$e = X\theta - y = \sum_{i=1}^{n} x_i^2$

OLS Objective

$= \|e\|^2 = e^\top e$

WLS Objective:

$= e^\top W e$

$= \sum_{i=1}^{n} w_i e_i^2$

b) $\hat{\theta}_{OLS} = (X^T X)^{-1} X^T Y$

$\hat{\theta}_{WLS} = (X^T W X)^{-1} X^T W Y$

c) $y_i = X_i^T \theta + \epsilon_i$

$y_i - x_i^T \theta = \epsilon_i \sim N(0, \sigma_i^2)$

$\Rightarrow \hat{\theta}_{MLE} = \underset{\theta}{\operatorname{argmax}} \; P(y_1, \dots, y_n \mid x_1, \dots, x_n; \theta)$

$= \underset{\theta}{\operatorname{argmax}} \prod_{i=1}^{n} P(Y = y_i \mid X = x_i; \theta)$

$= \underset{\theta}{\operatorname{argmax}} \prod_{i=1}^{n} P(\epsilon_i = y_i - x_i^T \theta)$?

$= \sum_{i=1}^{n} \frac{(y_i - x_i^T \theta)^2}{2\sigma_i^2}$

$w_i = \frac{1}{\sigma_i^2}$ or $\frac{1}{2\sigma^2}$.

4 Ridge Regression vs PCA (12 pts)

Assume we are given n training data points $(\mathbf{x}_i, y_i)$. We collect the responses into $\mathbf{y} \in \mathbb{R}^n$, and the features into the matrix $\mathbf{X} \in \mathbb{R}^{n \times d}$ where the rows are the $d-$dimensional feature vectors $\mathbf{x}_i^\top$ corresponding to each training point. Furthermore, assume that $\frac{1}{n}\sum_{i=1}^n \mathbf{x_i} = \mathbf{0}$, $n > d$ and $\mathbf{X}$ has rank d.

In this problem we want to compare two procedures: The first is ridge regression with hyper-parameter λ, while the second is applying ordinary least squares after using PCA to reduce the feature dimension from d to k (we give this latter approach the short-hand name k-PCA-OLS where k is the hyper-parameter).

Notation: The singular value decomposition of $\mathbf{X}$ reads $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$ where $\mathbf{U} \in \mathbb{R}^{n \times n}$, $\mathbf{\Sigma} \in \mathbb{R}^{n \times d}$ and $\mathbf{V} \in \mathbb{R}^{d \times d}$. We denote by $\mathbf{u}_j$ the n-dimensional column vectors of $\mathbf{U}$ and by $\mathbf{v}_j$ the $d-$dimensional column vectors of $\mathbf{V}$. Furthermore the diagonal entries $\sigma_j = \Sigma_{j,j}$ of $\mathbf{\Sigma}$ satisfy $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_d > 0$. For notational convenience, assume that $\sigma_j = 0$ for $j > d$.

(a) (4 pts) It turns out that the ridge regression optimizer (with $\lambda > 0$) in the $\mathbf{V}$-transformed coordinates

$$\widehat{\theta}_{\text{ridge}} = \arg\min_\theta \|\mathbf{X}\mathbf{V}\theta - \mathbf{y}\|_2^2 + \lambda\|\theta\|_2^2$$

has the following expression:

$$\widehat{\theta}_{\text{ridge}} = \text{diag}\left(\frac{\sigma_j}{\lambda + \sigma_j^2}\right)\mathbf{U}^\top\mathbf{y}. \tag{4}$$

Use $\widehat{y}_{test} = \mathbf{x}_{test}^\top \mathbf{V}\widehat{\theta}_{\text{ridge}}$ to denote the resulting prediction for a hypothetical $\mathbf{x}_{test}$. Using (4) and the appropriate scalar $\{\alpha_i\}$, this can be written as:

$$\widehat{y}_{test} = \mathbf{x}_{test}^\top \sum_{j=1}^d \mathbf{v}_j \alpha_j \mathbf{u}_j^\top \mathbf{y}. \tag{5}$$

What are the $\alpha_j \in \mathbb{R}$ for this to correspond to (4) from ridge regression? Show your work.

$\hat{y}_{test} = x_{test}^T V \hat{\theta}_{ridge}$

$= x_{test}^T \sum_{j=1}^{d} v_j \alpha_j u_j^T y$

$\hat{\theta}_{ridge} = \begin{pmatrix} \frac{\sigma_1}{\sigma_1^2+\lambda} & & \\ & \ddots & \\ & & \frac{\sigma_d}{\sigma_d^2+\lambda} \end{pmatrix}$

$\alpha_j = ?$

$V\hat{\theta}_{ridge} = \sum_{j=1}^{d} v_j \alpha_j v_j^T y$

$= V\tilde{D}U^T y$

$= \begin{bmatrix} v_1 & v_2 & \cdots & v_d \end{bmatrix} \begin{bmatrix} \frac{\sigma_1}{\sigma_1^2+\lambda} & & \\ & \ddots & \\ & & \frac{\sigma_d}{\sigma_d^2+\lambda} \end{bmatrix} \begin{bmatrix} u_1^T y \\ \vdots \\ u_d^T y \end{bmatrix}$

$$= \begin{bmatrix} v_1 & v_2 & \cdots & v_d \end{bmatrix} \begin{bmatrix} \frac{\sigma_1}{\sigma_1^2+\lambda} u_1^T y \\ \vdots \\ \frac{\sigma_d}{\sigma_d^2+\lambda} u_d^T y \end{bmatrix} = \sum_{j=1}^{d} v_j \frac{\sigma_j}{\sigma_j^2+\lambda}$$

$$\alpha_j = \frac{\sigma_j}{\sigma_j^2+\lambda}$$

$$\hat{\theta}_{Ridge} : \alpha_j = \frac{\sigma_j}{\sigma_j^2+\lambda}$$

$$\hat{\theta}_{OLS} : \alpha_j \Big|_{d=0} = \frac{\sigma_j}{\sigma_j^2} = \frac{1}{\sigma_j}$$

(b) (5 pts) Suppose that we do k-PCA-OLS — i.e. ordinary least squares on the reduced k-dimensional feature space obtained by projecting the raw feature vectors onto the $k < d$ principal components of the covariance matrix $\mathbf{X}^\top \mathbf{X}$. Use $\widehat{y}_{test}$ to denote the resulting prediction for a hypothetical $\mathbf{x}_{test}$,

It turns out that the learned k-PCA-OLS predictor can be written as:

$$\widehat{y}_{test} = \mathbf{x}_{test}^\top \sum_{j=1}^{d} \mathbf{v}_j \alpha_j \mathbf{u}_j^\top \mathbf{y}. \tag{6}$$

Give the $\alpha_j \in \mathbb{R}$ coefficients for k-PCA-OLS. Show work. These α_j's would be different compared to the previous part.

Hint 1: some of these α_j will be zero. Also, if you want to use the compact form of the SVD, feel free to do so if that speeds up your derivation.

Hint 2: some inspiration may be possible by looking at the next part for an implicit clue as to what the answer might be.

$$\hat{y}_{test} = x_{test}^T \sum_{j=1}^{d} V_j \alpha_j U_j^T y$$

$\alpha_j = ?$ $\begin{cases} \frac{1}{\sigma_j} & \text{for } j \le k \\ 0 & \text{for } j > k. \end{cases}$

Top K PCs are corresponding to $\sigma_1, \ldots, \sigma_k$ and in V-Space, $\hat{\theta}_j$ is <u>independent</u> of $\hat{\theta}_j$ for $i \neq j$.

$V = [V_1 \cdots V_k | V_{k+1} \cdots V_d]$

V_k V_{n-k}

$$\min \|XV\theta - y\|_2^2 \qquad X = U\Sigma V^T$$

$$V = [v_{1s}, \dots, v_{rs}, \dots v_d] \qquad X^TX = V\Sigma^2V^T \qquad V_k^T = V$$

$$\hat{\theta} = (X^TXV_r)^{-1}V_r^TX^Ty$$

$$\hat{\theta} = (V_r^TU\Sigma^2V^TV_r)^{-1}V_k^TV\Sigma U^Ty \qquad = [V_k \; 0]$$

$$= ([I_r\ 0]\begin{pmatrix}\sigma_1^2 & & \\ & \ddots & \\ & & \sigma_r^2\end{pmatrix}\begin{bmatrix}I_r \\ 0\end{bmatrix})^{-1}[I_r\ 0]\begin{pmatrix}\sigma_1 & & \\ & \ddots & \\ & & \sigma_d\end{pmatrix}U^Ty$$

$$y_{res} = X_{res}V_k\hat{\theta}$$

$$V_k^TV = V_k^T[V_k \quad V_{-k}]$$

$$= [V_r^TV_k \quad V_r^TV_r] = [I_r, 0]$$

$$\begin{pmatrix}\sigma_1^2 & & & \\ & \ddots & & \\ & & \sigma_r^2 & \\ & & & 0\end{pmatrix}\begin{pmatrix}I_r \\ 0\end{pmatrix} = \begin{pmatrix}\sigma_1^2 & & \\ & \ddots & \\ & & \sigma_r^2\end{pmatrix}$$

$$(A\ B)\begin{pmatrix}C \\ D\end{pmatrix} = AC + BD$$

$$= \begin{pmatrix}\sigma_1^2 & & \\ & \ddots & \\ & & \sigma_r^2\end{pmatrix}^{-1}\begin{pmatrix}\sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & 0\end{pmatrix}\begin{bmatrix}u_1^Ty \\ \vdots \\ u_d^Ty\end{bmatrix}$$

$$* = \begin{pmatrix} \frac{1}{\sigma_1} & & & \\ & \ddots & & 0 \\ & & \frac{1}{\sigma_k} & \end{pmatrix} \begin{pmatrix} u_1^T y \\ \vdots \\ u_d^T y \end{pmatrix} = \begin{pmatrix} u_1^T y / \sigma_1 \\ u_2^T y / \sigma_2 \\ \vdots \\ u_k^T y / \sigma_k \end{pmatrix}$$

$$V_k \hat{\theta} = \begin{bmatrix} v_1, \ldots, v_k \end{bmatrix} \begin{bmatrix} u^T y / \sigma_1 \\ \vdots \\ u^T y / \sigma_k \end{bmatrix}$$

$$= \sum_{j=1}^{k} v_j \frac{1}{\sigma_j} u_j^T y = X_{train} \sum_{j=1}^{d} v_j d_j u_j^T y.$$

Redo Sample Midterm:

1. a) True - Predictions are only meaningful if our data is representative of current & future population.

1. b) False. Its rank = k

$$HY = \hat{Y} = X\hat{\beta} = X(X^TX)^{-1}X^TY$$

$$H = X(X^TX)^{-1}X^T$$

1. c) False - Local minimizer

1. d) False - $\nabla_\theta(\theta^TA\theta) = \theta^TA^T + \theta^TA \neq 2A\theta$

1. e) True - Linearly dependent → Nx not full rank

1. f) True - See some betas to 0

1. g) True -

1. h) False - No relationship to Ridge

1. i) False - PCA is unsupervised learning → not used for prediction

1. j) True - only scale when units are different

2. a)

$$P(X) = \frac{P(X, Z)}{P(Z)} \qquad \sum_{i=1}^{2} P(Z = z_i) = 1$$

$$= \sum_{i=1}^{n} \sum_{j=1}^{2} P(X = x_i, Z = z_j)$$

$$= \sum_{i=1}^{n} \left[P(X = x_i, Z = 1) + P(X = x_i, Z = 2) \right]$$

$$= \sum_{i=1}^{n} \left[P(X = x_i \mid Z = 1) P(Z = 1) + P(X = x_i \mid Z = 2) P(Z = 2) \right]$$

$$= \sum_{i=1}^{n} \left[e^{-x_i} \frac{\lambda^{x_i}}{(x_i)!} w + e^{-x_i} \frac{\mu^{x_i}}{(x_i)!} (1 - w) \right]$$

$$= \sum_{i=1}^{n} \frac{e^{-x_i}}{(x_i)!} \left(\lambda^{x_i} w + \mu^{x_i} (1 - w) \right)$$

$$\Rightarrow \sum_{i=1}^{n} \frac{e^{-x_i}}{(x_i)!} \left(\lambda^{x_i} w + \mu^{x_i} (1 - w) \right)$$

$$\Rightarrow \sum_{i=1}^{n} -x_i + \log\left(\frac{\lambda^{x_i}}{(x_i)!} w \right) - x_i - \log\left(\frac{\mu^{x_i}}{(x_i)!} (1 - w) \right)$$

3. a) $\min_{\theta} \sum_{i=1}^{n} w_i (x_i^T \theta - y_i)^2$

$= \min_{\theta} \left(w_1 (x_1^T \theta - y_1)^2 + w_2 (x_2^T \theta - y_2)^2 + \cdots + w_n (x_n^T \theta - y_n)^2 \right)$

$= \min_{\theta} \left[W \| X\theta - y \|_2^2 \right]$ $\quad (X^T W X)^{-1} X^T W y$

$= \min_{\theta} \left[(X\theta - y)^T W (X\theta - y) \right]$

3.b) $\min_{\theta} \left[(X\theta - y)^T W (X\theta - y) \right]$

$\nabla_\theta \left(\theta^T X^T W X \theta - \theta^T X^T W y - y^T W X \theta - y^T W y \right)$

$= \theta^T X^T W X + \theta^T X^T W X - y^T W^T X - y^T W X \Big)$

$= 2\theta^T X W X - 2 y^T W^T X = 0$

$\theta^T X W X = y^T W^T X$

3.a) $\min_{\theta} \left[w \| X\theta - y \|_2^2 \right]$

$= \min_{\theta} \left[(X\theta - y)^T w \ (X\theta - y) \right]$

$\Rightarrow \nabla_\theta \left[\theta^T X^T w X \theta - \theta^T X^T w y - y^T w X \theta + y^T w y \right]$

$= 2\theta^T X^T w X - y^T w^T X - y^T w X \Big] = 0$

$2\theta^T X^T w X = y^T (w^T + w) X$

$2\theta^T X^T w X = 2 y^T w X$

$\theta^T = (X^T w X)^{-1} y^T w X$

$\theta = (X^T w X)^{-1} X^T w y$

3.c) $y_i = x_i^T \theta + \varepsilon_i$

Maximum Likelihood

$$\underset{\theta}{\operatorname{argmax}}\ P(y_1 = y_1, y_2 = y_2 \ldots \mid x_1 = x_1, x_2 = x_2 \ldots)$$

$$= \underset{\theta}{\operatorname{argmax}}\ P(\varepsilon_1 = y_1 - x_1^T\theta;\ \varepsilon_2 = y_2 - x_2^T\theta \ldots)$$

$$= \underset{\theta}{\operatorname{argmax}} \left[\frac{1}{\sqrt{\det(2\pi\sigma^2)}} e^{\left(-\frac{(y_1 - x_1^T\theta)^2}{2\sigma_i^2}\right)} * \ldots \right]$$

$$= \underset{\theta}{\operatorname{argmax}} \left[\sum_{i=1}^{n} - \frac{(y_i - x_i^T\theta)^2}{2\sigma_i^2} \right]$$

$$= \underset{\theta}{\operatorname{argmin}} \left[\sum_{i=1}^{n} w_i (x_i^T\theta - y_i)^2 \right]$$

$$w_i = \frac{1}{2\sigma_i^2}$$

4.a)

$$\hat{y}_{test} = x_{test}^T V \hat{\theta}_{ridge} = x_{test}^T \sum_{j=1}^{d} v_j \alpha_j u_j^T y$$

$$V\hat{\theta}_{ridge} = V \, diag\left(\frac{\sigma_j}{\lambda + \sigma_i^2}\right) U^T y$$

$$\sum_{j=1}^{d} v_j \alpha_j u_j^T y = v_1 \alpha_1 u_1^T y + v_2 \alpha_2 u_2^T y + \cdots + v_d \alpha_d u_d^T y$$

$$= \left(\sum_{j=1}^{d} v_1 \alpha_1\right) U^T y$$

$$= V\left(\sum_{j=1}^{d} \alpha_i\right) U^T y$$

$$\alpha_1 + \alpha_2 + \cdots + \alpha_d = diag\left(\frac{\sigma_j}{\lambda + \sigma_j^2}\right)$$

4.a)

$$\hat{y}_{test} = x^T_{test} V \hat{\theta}_{ridge}$$

$$V\hat{\theta}_{ridge} = V \underset{d \times d}{\text{diag}}\left(\frac{\sigma^2}{\sigma^2 + \lambda}\right) \underset{d \times n}{U^T} Y$$

$$V\hat{\theta}_{ridge} = V \begin{bmatrix} \frac{\sigma^2}{\sigma^2+\lambda} & & \\ & \ddots & \\ & & \frac{\sigma^2}{\sigma^2+\lambda} \end{bmatrix}$$

4 Ridge Regression vs PCA (12 pts)

Assume we are given n training data points $(\mathbf{x}_i, y_i)$. We collect the responses into $\mathbf{y} \in \mathbb{R}^n$, and the features into the matrix $\mathbf{X} \in \mathbb{R}^{n \times d}$ where the rows are the $d-$dimensional feature vectors $\mathbf{x}_i^\top$ corresponding to each training point. Furthermore, assume that $\frac{1}{n}\sum_{i=1}^n \mathbf{x_i} = \mathbf{0}$, $n > d$ and $\mathbf{X}$ has rank d.

In this problem we want to compare two procedures: The first is ridge regression with hyper-parameter λ, while the second is applying ordinary least squares after using PCA to reduce the feature dimension from d to k (we give this latter approach the short-hand name k-PCA-OLS where k is the hyper-parameter).

Notation: The singular value decomposition of $\mathbf{X}$ reads $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$ where $\mathbf{U} \in \mathbb{R}^{n \times n}$, $\mathbf{\Sigma} \in \mathbb{R}^{n \times d}$ and $\mathbf{V} \in \mathbb{R}^{d \times d}$. We denote by $\mathbf{u}_j$ the n-dimensional column vectors of $\mathbf{U}$ and by $\mathbf{v}_j$ the $d-$dimensional column vectors of $\mathbf{V}$. Furthermore the diagonal entries $\sigma_j = \Sigma_{j,j}$ of $\mathbf{\Sigma}$ satisfy $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_d > 0$. For notational convenience, assume that $\sigma_j = 0$ for $j > d$.

(a) (4 pts) It turns out that the ridge regression optimizer (with $\lambda > 0$) in the $\mathbf{V}$-transformed coordinates

$$\widehat{\theta}_{\text{ridge}} = \arg\min_\theta \|\mathbf{X}\mathbf{V}\theta - \mathbf{y}\|_2^2 + \lambda\|\theta\|_2^2$$

has the following expression:

$$\widehat{\theta}_{\text{ridge}} = \text{diag}(\frac{\sigma_j}{\lambda + \sigma_j^2})\mathbf{U}^\top\mathbf{y}. \tag{4}$$

Use $\widehat{y}_{test} = \mathbf{x}_{test}^\top \mathbf{V}\widehat{\theta}_{\text{ridge}}$ to denote the resulting prediction for a hypothetical $\mathbf{x}_{test}$. Using (4) and the appropriate scalar $\{\alpha_i\}$, this can be written as:

$$\widehat{y}_{test} = \mathbf{x}_{test}^\top \sum_{j=1}^d \mathbf{v}_j \alpha_j \mathbf{u}_j^\top \mathbf{y}. \tag{5}$$

What are the $\alpha_j \in \mathbb{R}$ for this to correspond to (4) from ridge regression? Show your work.

$$x_{test}^\top V\hat{\theta}_{ridge} = x_{test}^\top \sum_{j=1}^{d} v_j \alpha_j u_j^\top y$$

$$V\hat{\theta}_{ridge} = \sum_{j=1}^{d} v_j \alpha_j u_j = v_1\alpha_1 u_1 + v_2\alpha_2 u_2 + \cdots + v_d\alpha_d u_d$$

$$\begin{bmatrix} | & | & & | \\ v_1 & v_2 & \cdots & v_d \\ | & | & & | \end{bmatrix} \begin{bmatrix} \frac{\sigma_1}{\lambda+\sigma_1^2} & & \\ & \ddots & \\ & & \frac{\sigma}{\lambda+\sigma_j^2} \end{bmatrix} \begin{bmatrix} — & u_1 & — \\ — & u_2 & — \\ & \vdots & \\ — & u_n & — \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_n \\ \vdots \\ y_n \end{bmatrix}$$

$d \times d$ $\quad d \times n$ $\quad n \times n$ $\quad n \times 1$

$$\begin{bmatrix} & & \\ & \text{diag}\begin{pmatrix} a_1 & & \\ & a_2 & \\ & & a_3 \end{pmatrix} & \begin{pmatrix} a_1 & & 0 \\ & a_2 & 0 \\ & & a_3 \end{pmatrix} \end{bmatrix}$$

$$V\hat{\theta}_{ridge} = \sum_{j=1}^{d} v_j \alpha_j u_j y$$

$$LHS = V\hat{\theta}_{ridge}$$

$$= V \, diag\left(\frac{\sigma_j}{\sigma_j^2 + \lambda}\right) U^T y$$

$$= \underbrace{\begin{bmatrix} | & | & & | \\ v_1 & v_2 & \cdots & v_d \\ | & | & & | \end{bmatrix}}_{V} \begin{bmatrix} \frac{\sigma_1}{\sigma_1^2+\lambda} & & & 0 \\ & \ddots & & \\ & & \frac{\sigma_j}{\sigma_j^2+\lambda} & \\ 0 & & & 0 \end{bmatrix} \begin{bmatrix} — & u_1 & — \\ — & u_2 & — \\ & \vdots & \\ — & u_n & — \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

$$RHS = \sum_{j=1}^{d} v_j \alpha_j u_j$$

$$= v_1 \alpha_1 u_1 y + v_2 \alpha_2 u_2 y + \cdots$$

$$=$$

$$= v_1 \cdot \frac{\sigma_1}{\sigma_1^2+\lambda} \cdot u_1 y_1$$

$$\alpha_i = \frac{\sigma_i}{\sigma_i^2 + \lambda}$$

$n \times n \quad n \times n \quad n \times 1$

$d \times 1$

$\text{diag}(\quad) \; U^T y$

$n \times 1$

$d < n$

$V \hat{\theta}_{\text{ridge}} = d \times d \quad n \times 1$

$X = U \Sigma V^T$

$n \times n \quad n \times d \quad d \times d$

(b) (5 pts) Suppose that we do k-PCA-OLS — i.e. ordinary least squares on the reduced k-dimensional feature space obtained by projecting the raw feature vectors onto the $k < d$ principal components of the covariance matrix $\mathbf{X}^\top \mathbf{X}$. Use $\widehat{y}_{test}$ to denote the resulting prediction for a hypothetical $\mathbf{x}_{test}$,

It turns out that the learned k-PCA-OLS predictor can be written as:

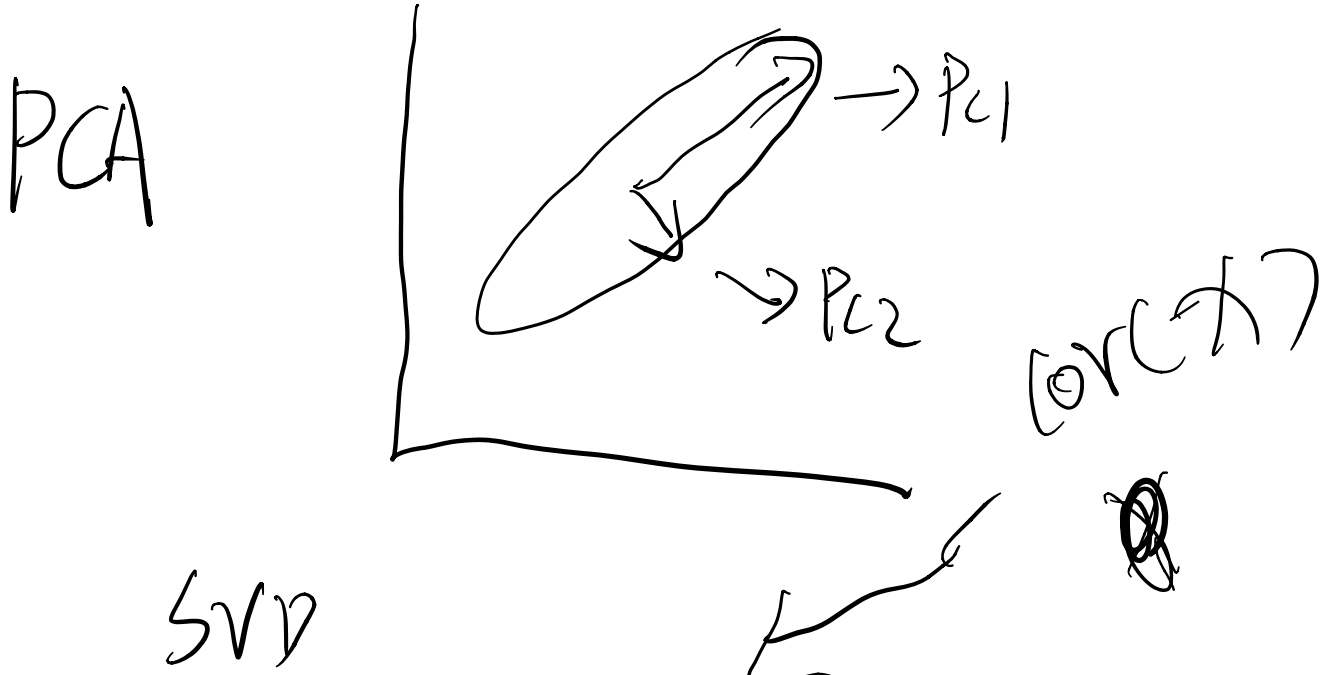
$$\widehat{y}_{test} = \mathbf{x}_{test}^\top \sum_{j=1}^{d} \mathbf{v}_j \alpha_j \mathbf{u}_j^\top \mathbf{y}. \tag{6}$$

Give the $\alpha_j \in \mathbb{R}$ coefficients for k-PCA-OLS. Show work. These α_j's would be different compared to the previous part.

Hint 1: some of these α_j will be zero. Also, if you want to use the compact form of the SVD, feel free to do so if that speeds up your derivation.

Hint 2: some inspiration may be possible by looking at the next part for an implicit clue as to what the answer might be.

4.b) $\hat{y}_{test} = x_{test}^T \sum_{j=1}^{d} v_j \alpha_j u_j^T y$



$X = U \Sigma V^T$ $X^T X = V \Sigma^2 V^T$

Σ ← eigenvalue

$\Sigma = XV$

scores aka principal components

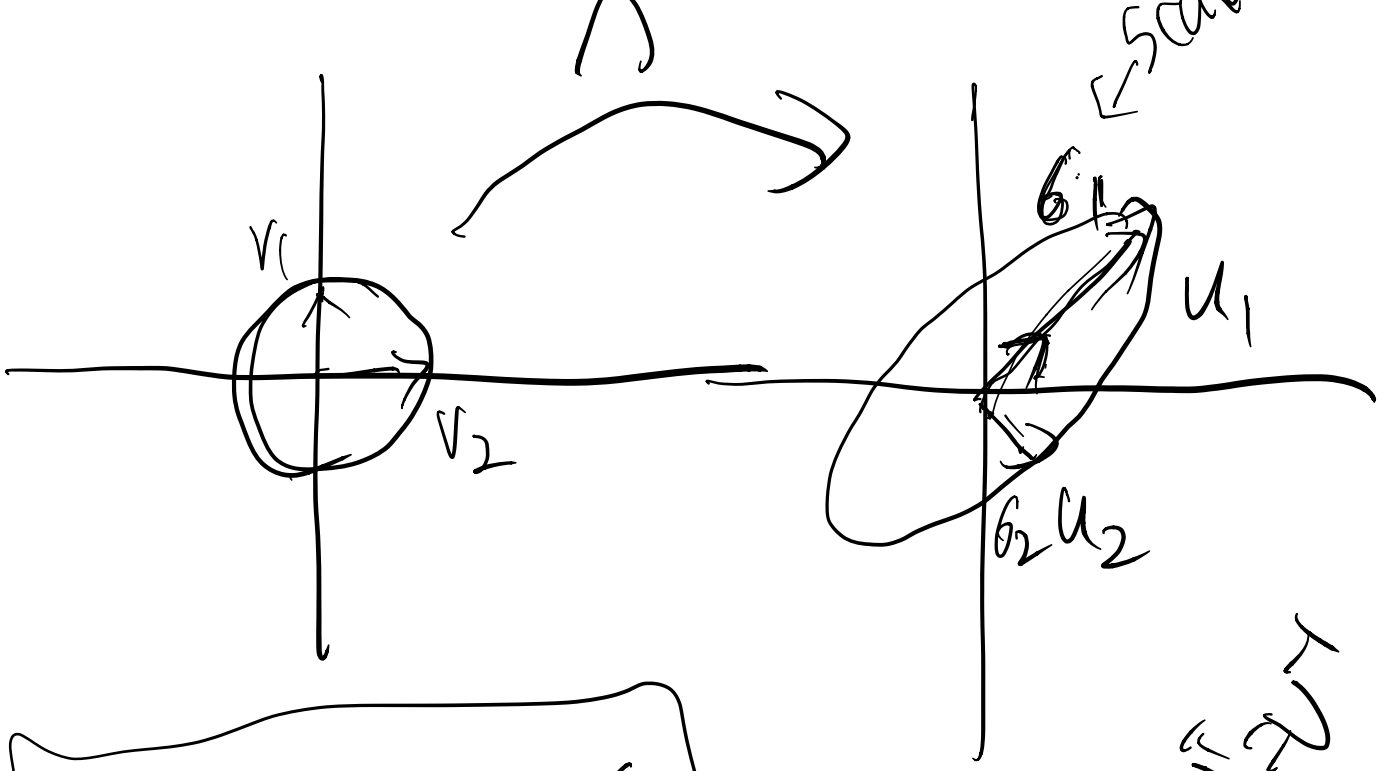
V is called rotation matrix → loadings

its eigenvectors of $X^T X$ → primal vectors

$Xa = b$

v_1 unit vector

u_1



$$XV = U\Sigma$$

$[v_1, v_2]$

$X = U\Sigma V^T$

$UU^T = I$

$U^T U = I$

$(x_1 \cdots x_n)$

$\begin{pmatrix} x_1^T \\ \vdots \\ x_n^T \end{pmatrix}$

σ_1

Σ =
σ1 ~ σn
OLS
Xr
X'X
z1 z2 z3
d=10
K=10

$\hat{y} = X\hat{\beta}$

$\|Y - XV\beta\|_2^2 = L(\theta)$

P_{PCA}

$\hat{\beta}_{OLS} = (X^T X)^{-1} X^T Y$

$(V^T X^T X V)^{-1} V^T X^T Y$

$= (V^T V \Sigma^2 V^T V)^{-1} V^T X^T Y$

$X^T X$: $(n \times d)(d \times n)(n \times d)(d \times d)$

$X = U \Sigma V^T$

$X^T X = V \Sigma^T U^T U \Sigma V^T$

$X^T X = V \Sigma^2 V^T$

$= (\Sigma^2)^{-1} V^T X^T Y$

$= V$

$= (\Sigma)^{-1} \Sigma^T U^T Y$

$= (\Sigma)^{-1} \Sigma^T U^T Y$

$n \times d \quad d \times d$

$\begin{bmatrix} \sigma_{11} & & \\ & \ddots & \\ & & \sigma_{dd} \end{bmatrix} \begin{bmatrix} -v_1- \\ -v_2- \\ \vdots \\ -v_d- \end{bmatrix} \begin{bmatrix} & & \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$

3.6) $\min_{\theta} \|\sqrt{W} X\theta - y\|_2^2$

$$\hat{\theta}_{ols} = (X^T X)^{-1} X^T Y$$

$$= (X^T \sqrt{W}^T \sqrt{W} X)^{-1} \sqrt{W} X^T Y$$

$$= (X^T W X)^{-1} \sqrt{W} X^T Y$$

$$(\Sigma)^{-1} \Sigma^T U^T Y = \sum u_j \alpha_j u_j^T y$$

$\Sigma^T = \Sigma^{-1}$

$$\begin{bmatrix} \frac{1}{\sigma_1^2} & & & 0 \\ & \frac{1}{\sigma_2^2} & & \\ & & \ddots & \\ 0 & & & \frac{1}{\sigma_d^2} \end{bmatrix} \begin{bmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ 0 & 0 & 0 & \sigma_d \end{bmatrix} \begin{bmatrix} \end{bmatrix}$$

$$\begin{bmatrix} \frac{1}{\sigma_1} & & & 0 \\ & \frac{1}{\sigma_2} & & \\ & & \ddots & \\ 0 & & & \frac{1}{\sigma_d} \end{bmatrix} = \begin{bmatrix} | & | & & | \\ u_1 & u_2 & \cdots & u_n \\ | & | & & | \end{bmatrix} \alpha,$$

$$\alpha_j = \frac{1}{\sigma_j} u_j \qquad \alpha_0 = \frac{1}{\sigma_0}$$

$\hat{y}$

$$X V \hat{\beta} = X V \alpha U^T Y$$

$$X V (\Sigma^{-2}) \Sigma^T U^T Y = X V \alpha U^T Y$$

$$\alpha = (\Sigma^{-2}) \Sigma^T$$

$n \times d$

$$\alpha = \begin{bmatrix} \frac{1}{\sigma_1^2} & & & \\ & \frac{1}{\sigma_2^2} & & \\ & & \ddots & \\ & & & \frac{1}{\sigma_d^2} \\ 0 & \cdots & \cdots & 0 \end{bmatrix} \begin{pmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_d \\ 0 & \cdots & \cdots & 0 \end{pmatrix}^{d \times n}$$

$\alpha \in \mathbb{R}^{}$

$$\alpha_{ii} = \frac{1}{\sigma_i}$$

$k < d$

$V = [v_1 \cdots v_k \; 0 \; 0]$

$$\hat{\beta}_{PLA} = X V \Sigma^{-2} \Sigma^{T} U^{T} Y$$

$n \times n \quad n \times d \quad d \times n$

$$= \begin{pmatrix} \frac{1}{\sigma_1^2} & & \\ & \ddots & \\ & & \frac{1}{\sigma_j^2} \end{pmatrix} \begin{pmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_j \end{pmatrix}$$

$n \times n$

$$= X V \begin{pmatrix} \frac{1}{\sigma_1} & & \\ & \ddots & \\ & & \frac{1}{\sigma_j} \end{pmatrix} U^T Y$$

$$= X_{tese} \sum_{i=1}^{d} v_i \left(\frac{1}{\sigma_i}\right) u_i^T y$$

$i = 0 \Rightarrow i = k$: $\frac{1}{\sigma_i}$

$i > k$: 0