- Categorical, or nominal
  - Property: distinctness
  - Examples: ID numbers, eye color, zip code
- Ordinal
  - Property: distinctness, order
  - Examples: pain on a scale 1-10, t-shirt size (XS, S, M, L, XL)
- Interval: has values in equal intervals
  - Property: distinctness, order, addition
  - Examples: calendar dates
- Ratio (can say three days later but doesn't make sense to say three times later)
  - Property: distinctness, order, addition, multiplication
  - Examples: distance, speed, weight

Data matrix: n observations (rows), p variables (columns)

- Record data
- Document data
- Graph data
- Spatial data
- Temporal or sequential data

- Data that consists of a collection of records, each of which consists of a fixed set of variables
- Data matrix conversion: simply arrange records in rows
- Term: words, usually stripped of endings and common "stop words" (e.g. go, going, goes → go, no "the", "and", etc)
- Variables: terms
- Observations: documents
- Value of the variable: the number of times the corresponding term occurs in the document (often normalized)
- A special type of record data
- Each record (transaction) involves a set of items
- Example: a grocery store purchase constitutes a transaction, and the individual products purchased are the items.
- Consists of objects (nodes) and connections between them (edges)
  - Internet (the Web, social networks)
  - Computer / mobile / electric grid networks
  - Transportation
  - Ecosystems (predator / prey networks)
- Spatial data (e.g. temperature at various weather stations in the US)
- Temporal data (time series): (e.g. stock prices over a year)
- Functional data (e.g. spectral measurements at different wavelengths)
- Sequential data (e.g. human genome)

- Supervised learning: predicting an outcome
- Regression: predicting a continuous outcome
- Classification: predicting a categorical outcome

Why not just always use a more flexible method?

- A simple method such as linear regression is much easier to interpret (inference). For example, in linear regression β_j is the average increase in y for a one unit increase in x_j holding all other variables constant.
- Even for prediction purposes, a simple model can be more accurate if there are not enough data points to fit a more flexible model
- Overfitting: too much flexibility follows the noise too closely

Goodness of Fit: R^2

- Some of the variation in y can be explained by changes in x's and some cannot.
- Total variation (Total Sum of Squares): $\sum_{i=1}^n (y_i - \bar{y})^2$
- Unexplained variation (Residual Sum of Squares): $\sum_{i=1}^n \hat{e}_i^2$
- R^2: the fraction of variance "explained" by x.

$$R^2 = 1 - \frac{\text{RSS}}{\sum(y_i - \bar{y})^2}$$

- R^2 is always between 0 and 1.
- $R^2 = 0$ means no variance in y is explained by x. $R^2 = 1$ means perfect fit to the data ($\hat{y}_i = y_i$ for all i).
- Hypothesis testing framework: assume x_j is not useful ($\beta_j = 0$) and see if there is enough evidence to reject this hypothesis.
- $H_0 : \beta_j = 0$ vs $H_a : \beta_j \neq 0$
- Because $\hat{\beta}$ is approximately normal, t-test applies: calculate the t-statistic

$$t = |\hat{\beta}_j| / \text{SD}(\hat{\beta}_j)$$

- If t is large (equivalently p-value is small) we can reject $H_0 : \beta_j = 0$ and conclude x_j is useful in this model.

- Need a hypothesis test for
  - H_0: at $\beta_1 = \beta_2 = \cdots = \beta_p = 0$ against
  - H_a: at least one $\beta_j \neq 0$
- Tested by the F-test in ANOVA (ANalysis Of VAriance) table.

$$F = \frac{(\text{TSS} - \text{RSS})/p}{\text{RSS}/(n-p-1)}$$

	df	SS	MS	F-value	p-value
Explained	2	4860.2	2430.1	859.6	0.000
Unexplained	197	556.9	2.83		

- Non-linearity of the data
- Dependence among errors
- Non-constant variance of error terms
- Outliers
- High leverage points
- Collinearity ??

\hat{\beta}=(X^TX)^{-1}X^TY 如果X矩阵的列向量x1,x2,x3是highly colinear的 那么X^TX很可能rank就会变小 然后就没有inverse了 或者inverse变得很大

K-nearest neighbors can fail in high dimensions, because it becomes difficult to find K observations close to a target point x_0:

- Points in high dimensions are far apart: finding K neighbors means taking a large spatial neighborhod. (Increased bias).
- Reducing the spatial size of the neighborhood means reducing K; the predictions become noisier (Increased variance).

- Volume of the large unit cube: 1^d=1
- Volume of the small cube with edge ℓ: l^d
- Fraction of observations that fall in the cube: l^d/1
- For a given fraction ρ, need a cube with edge length l=p^(1/d)

dimension p, fraction ρ

- When $p = 1$: If $\rho = 0.01$, $\ell = 0.01$ and if $\rho = 0.1$, $\ell = 0.1$.
- When $p = 10$: If $\rho = 0.01$, $\ell = 0.63$ and if $\rho = 0.1$, $\ell = 0.80$.

When $p = 10$, in order to capture 10% of the data, we must cover 80% of the range of each input.

K-Nearest Neighbors (KNN) for Classification

- Classification setting: y is a categorical variable (class)
- For any given x we find the K closest neighbors to x in the training data, and examine their classes.
- Assign x to the class corresponding to the majority votes of the K nearest neighbors
- If a tie occurs, choose at random (K is usually taken to be odd to avoid ties for two classes)
- Code $Y = 1$ for class 1, and $Y = -1$ for class 2.
- Take a vote on a new point x:

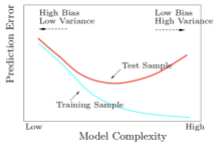
$$\hat{f}(x) = \frac{1}{K} \sum_{x_i \in N_K(x)} y_i$$

where $N_K(x)$ consists of the k closest points to x in the training data (k-nearest neighborhood).

- The classification rule is

$$\hat{c}(x) = \begin{cases} 1 & \text{if } \hat{f}(x) > 0 \\ -1 & \text{if } \hat{f}(x) < 0 \end{cases}$$

- But each neighborhood makes its own local estimate; there are n training points, K neighbors for each, and thus roughly n/K different local estimates
- K controls the model complexity: the smaller K, the more different local estimates. the more complex the model.
- In general, as the model complexity increases, training errors will always decline.
- However, test errors will decline at first (as reductions in bias dominate) but will then start to increase again (as increases in variance dominate).
- We must always keep this in mind when choosing a learning method. More flexible/complicated is not always better!



- Bias refers to systematic error introduced by approximating a real life problem by a model (e.g., a linear model).
- Formally, if $y = f(x) + \varepsilon$, $E\varepsilon = 0$, then

$$\text{bias}(\hat{f}(x)) = E\hat{f}(x) - f(x)$$

- The expectation is taken over the distribution of noise
- A method is called unbiased if $\text{bias}(x) = 0$ for all x
- In general, the more flexible a method, the lower its bias.
- Variance refers to random error resulting from sample variability; it measures how much $\hat{f}$ would change if you had a different training sample from the same distribution.
- Formally, if $y = f(x) + \varepsilon$, $E\varepsilon = 0$, then

$$\text{Var}(\hat{f}(x)) = E(\hat{f}(x) - E\hat{f}(x))^2$$

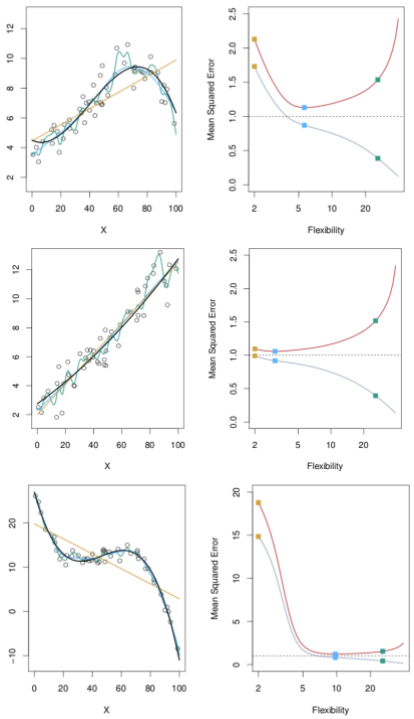
- The expectation is taken over the distribution of noise
- In general, the more flexible a method, the higher its variance.
- For a given fixed point x, the expected test MSE for a new y at x is

$$\begin{aligned} E(\text{MSE}(x)) &= E(y - \hat{f}(x))^2 \\ &= [E(\hat{f}(x)) - f(x)]^2 + E[\hat{f}(x) - E(\hat{f}(x))]^2 + \text{Var}(\varepsilon) \\ &= [\text{Bias}(\hat{f}(x))]^2 + \text{Var}(\hat{f}(x)) + \sigma^2 \end{aligned}$$

- Thus the expected test MSE may go up or down with increased complexity, depending on which term dominates.
- σ^2 is the irreducible noise; no method can do better than that.
- As long as $n/K > p$, KNN is more "flexible" than a linear model with p predictors.

$$\begin{aligned} Ey_{(\ell)} &= Ef(x_{(\ell)}) + E\varepsilon = f(x_{(\ell)}) \\ \text{Var}(y_{(\ell)}) &= \text{Var}(f(x_{(\ell)})) + \text{Var}(\varepsilon) = \sigma^2 \\ E\hat{f}(x) &= \frac{1}{K}\sum_{\ell=1}^{K} Ey_{(\ell)} = \frac{1}{K}\sum_{\ell=1}^{K} f(x_{(\ell)}) \\ \text{Var}(\hat{f}(x)) &= \frac{1}{K}\sum_{\ell=1}^{K} \text{Var}(y_{(\ell)}) = \frac{\sigma^2}{K} \\ E(\text{MSE}(x)) &= \left(f(x) - \frac{1}{K}\sum_{\ell=1}^{K} f(x_{(\ell)})\right)^2 + \frac{\sigma^2}{K} + \sigma^2 \end{aligned}$$

- The squared bias term tends to increase with K.
  - For small K, the closest neighbors have values $f(x_{(\ell)})$ similar to $f(x_0)$, at least if f is smooth.
  - For large K, "further away" points are counted as neighbors.
- The variance term decreases when K increases.



- Bayes optimal classifier:

$$\begin{aligned} C^*(x_0) &= \arg\min_C \text{R}(C) \\ &= \arg\max_k P(y = c_k | x = x_0) \end{aligned}$$

- The Bayes error rate is the error of the Bayes optimal classifier:

$$P(C^*(x) \neq y)$$

- This is the lowest possible error rate that can only be achieved if we knew exactly the "true" probability distribution of the data.
- Goal 1 (prediction): *new* data points (test set) should be assigned a class as accurately as possible.
- Goal 2 (inference): understand which of the variables help predict class, and how they are connected
- Let π_k be the prior probability of class k, $P(Y = c_k)$.
- Let $p_k(x)$ be the class-conditional density of X in class k.
- The posterior probability

$$P(Y = c_k | X = x) = \frac{p_k(x)\pi_k}{\sum_{k=1}^{K} p_k(x)\pi_k}$$

- The optimal classifier picks c_k such that $P(Y = c_k | X = x)$ is maximized
- Assume equal for all classes and drop them
- Estimate from data as

$$\hat{\pi}_k = \frac{n_k}{n}$$

where n_k is the number of observations from class k in the training data, $n = n_1 + n_2 + \cdots + n_K$.

- Use prior knowledge

Pros — Pros and cons of parametric methods

- The rule is fully determined by a small set of parameters
- Relatively easy to fit even in high dimensions
- Many quantities (e.g. error estimates) can be computed explicitly
- The most popular choice for $\hat{\theta}$ is the maximum likelihood estimator, lots of known good properties

Cons

- Parametric assumptions may not hold
- Even an excellent estimate of $\hat{\theta}$ does not always mean that $p(x;\hat{\theta})$ is a good approximation to $p(x;\theta)$ (requires extra conditions)

Pros — Pros and cons of nonparametric methods

- No need to make assumptions
- Very flexible, in principle can approximate any posterior probabilities

Cons

- Usually nothing can be computed explicitly
- Danger of overfitting (unless regularized)
- "Curse of dimensionality" – density estimation fails in high dimensions

- Let X be univariate ($p = 1$), $K = 2$, $p_1(x) \sim N(\mu_1, \sigma_1^2)$, $p_2(x) \sim N(\mu_2, \sigma_2^2)$

$$p_k(x) = \frac{1}{\sqrt{2\pi\sigma_k^2}} \exp\left(-\frac{1}{2\sigma_k^2}(x-\mu_k)^2\right)$$

- Compute the posterior probability: for $k = 1, 2$,

$$P(Y = k|x) = \frac{\pi_k p_k(x)}{\pi_1 p_1(x) + \pi_2 p_2(x)} \propto \pi_k p_k(x)$$

- Assume for simplicity $\pi_1 = \pi_2 = 0.5$. The Bayes rule assigns class 1 if $p_1(x) > p_2(x)$, class 2 otherwise
- Take logs and simplify
- The rule becomes: assign class 1 if   cross out if σ1=σ2

$$x^2\left(\frac{1}{\sigma_2^2} - \frac{1}{\sigma_1^2}\right) - 2x\left(\frac{\mu_2}{\sigma_2^2} - \frac{\mu_1}{\sigma_1^2}\right) + \left(\frac{\mu_2^2}{\sigma_2^2} - \frac{\mu_1^2}{\sigma_1^2}\right) + 2\log\left(\frac{\sigma_2}{\sigma_1}\right) > 0$$

otherwise assign class 2.
- This is a quadratic function in x ax+b>0
- What if we assume $\sigma_1 = \sigma_2$?   see page 22 change class once

- LDA: assume $\sigma_k^2 = \sigma^2$ for all k.   p=1,K>2
- Compare $P(Y = k|X = x) \propto p_k(x)\pi_k$
- For each k, the discriminant function is

Decision: max $\delta_k(x) = x\frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \log \pi_k$

Model each class density as multivariate Gaussian $N(\mu_k, \Sigma_k)$:

$$f_k(x) = \frac{1}{(2\pi)^{p/2}|\Sigma_k|^{1/2}} e^{-\frac{1}{2}(x-\mu_k)^\top \Sigma_k^{-1}(x-\mu_k)}.$$

y,x1,x2,x3
mu是一个向量
其中每一个值是x_n的均值

- μ_k: the mean vector for class k matrix_ij=cov(x_i,x_j), 对角线就是var(x_i)
- Σ_k: the covariance matrix for class k; $|\Sigma_k|$ is the determinant of Σ_k
- Each variable x_j is also marginally normal with mean μ_{kj} and variance $\Sigma_{k,jj}$.
- LDA: assume $\Sigma_k = \Sigma$ for all k.
- For each k, the discriminant function is

$$\delta_k(x) = x^\top \Sigma^{-1}\mu_k - \frac{1}{2}\mu_k^\top \Sigma^{-1}\mu_k + \log \pi_k$$

- Comparing class k and class k', the decision boundary is given by

$$\{x : x^\top \Sigma^{-1}(\mu_k - \mu_{k'}) + \log\frac{\pi_k}{\pi_{k'}} - \frac{1}{2}(\mu_k + \mu_{k'})^\top \Sigma^{-1}(\mu_k - \mu_{k'}) = 0\}$$

- Linear decision boundary, with the directional vector $\Sigma^{-1}(\mu_k - \mu_{k'})$; generally not in the direction of $(\mu_k - \mu_{k'})$.
- Assuming equal priors, we classify x to the class with the closest centroid to x, using the squared Mahalanobis distance corresponding to Σ:

$$(x - \mu_k)^T \Sigma^{-1}(x - \mu_k)$$

- Special case: $\Sigma = I$; then classify to the class with the closest centroid in Euclidean distance

$$(x - \mu_k)^T(x - \mu_k)$$

Parameter Estimation for LDA

In practice, we estimate parameters from training data.

- Class priors: let n_k be the number of training observations in class k,

$$\hat{\pi}_k = \frac{n_k}{n}$$

- Class means:

$$\hat{\mu}_k = \frac{1}{n_k}\sum_{i:y_i=c_k} x_i$$

- Class covariances:

$$\hat{\Sigma}_k = \frac{1}{n_k - 1}\sum_{i:y_i=c_k}(x_i - \hat{\mu}_k)(x_i - \hat{\mu}_k)^\top$$

- The pooled covariance

$$\hat{\Sigma} = \frac{1}{n-K}\sum_{k=1}^{K}\sum_{i:y_i=c_k}(x_i - \hat{\mu}_k)(x_i - \hat{\mu}_k)^\top$$

- Σ_k's are allowed to be different.   Quadratic discriminant analysis
- Discriminant function

$$\begin{aligned}\delta_k(x) &= -\frac{1}{2}\log|\Sigma_k| - \frac{1}{2}(x-\mu_k)^\top \Sigma_k^{-1}(x-\mu_k) + \log \pi_k \\ &= x^\top W_k x + x^\top w_k + b_k\end{aligned}$$

- The decision boundary between class k and class k' is a quadratic function

$$\{x : x^\top(W_k - W_{k'})x + x^\top(w_k - w_{k'}) + (b_k - b_{k'}) = 0\}$$

- QDA will work best when the variances are very different between classes and we have enough observations to accurately estimate the variances.
- LDA will work best when the variances are similar among classes or we don't have enough data to accurately estimate the variances.
- Assume independence among input variables when class is given

Naive Bayes Classifier

$$p_k(x_1, \ldots, x_p) = p_{k,1}(x_1)p_{k,2}(x_2)\ldots p_{k,p}(x_p)$$

- Estimate $p_{k,j}$ for each pair of k and j separately.
- New point is classified to c_k corresponding to the largest $\prod_{j=1}^{p}\hat{p}_{k,j}(x_j)\cdot\pi_k$.
- DQDA (diagonal QDA): assume each Σ_k is diagonal
- DLDA (diagonal LDA): assume the common Σ is diagonal
- Strong assumption, but often classifies well when p is large.

- Linear decision boundary, with the directional vector $\Sigma^{-1}(\mu_k - \mu_{k'})$; generally not in the direction of $(\mu_k - \mu_{k'})$.
- Assuming equal priors, we classify x to the class with the closest centroid to x, using the squared Mahalanobis distance corresponding to Σ:

$$(x - \mu_k)^T \Sigma^{-1}(x - \mu_k)$$

- Special case: $\Sigma = I$; then classify to the class with the closest centroid in Euclidean distance

$$(x - \mu_k)^T(x - \mu_k)$$

- Fisher: find linear combinations of variables $a'x$ such that between-class variance (the variance of class centroids) is maximized relative to within-class variance (pooled variance within classes)   alternative interpretation of LDA
- This interpretation does not require the normal assumption
- All of classification information for K classes is contained in $K - 1$ linear combinations; thus for classification purposes can just compute these $K - 1$ linear combinations and use LDA as a dimension reduction technique
- LDA and QDA are optimal under the normal assumption
- Work well in practice in many cases, even when the normal assumption is questionable
- Requires estimating a number of parameters proportional to p^2, so needs regularization when $p > n$
- Naive Bayes is a simple and effective regularization when p is much larger than n; there are others.
- The regression function $\beta_0 + \beta^\top x$ can take on any value between negative and positive infinity.
- In a classification problem, y can only take on two possible values: 0 or 1.   logistic regression (why not linear regression?)
- The point "cloud" has two y values only; the line is not a good model even in the range of x where y remains between 0 and 1

- Use the logit transformation (logistic function):

$$\log\frac{P(Y = c_1|x)}{P(Y = c_2|x)} = \beta_0 + \beta^\top x$$

- Can solve for probabilities:

$$\begin{aligned}P(Y = c_1|x) &= \frac{e^{\beta_0+\beta^\top x}}{1 + e^{\beta_0+\beta^\top x}} \\ P(Y = c_2|x) &= \frac{1}{1 + e^{\beta_0+\beta^\top x}}\end{aligned}$$

- The probabilities are automatically between 0 and 1, and $P(Y = c_1|x) + P(Y = c_2|x) = 1$.

Writing the likelihood of binary variables

- Would like to fit the model by maximum likelihood estimation
- Let $Y = 1$ with probability p or 0 with probability $1 - p$

$$P(Y = y) = \begin{cases} p, & \text{if } y = 1 \\ 1 - p, & \text{if } y = 0 \end{cases} = p^y(1-p)^{1-y}$$

- Observe an i.i.d. sample from this distribution: $y_1, y_2, \ldots, y_n$
- Likehood as a function of p:

$$L(p; y_1, \ldots, y_n) = \prod_{i=1}^{n} p^{y_i}(1-p)^{1-y_i}$$

- Log-likelihood:

$$\ell(p; y_1, \ldots, y_n) = \log L = \sum_{i=1}^{n} y_i \log p + (1 - y_i)\log(1-p)$$

- Write $\beta = (\beta_0, \beta)$
- Write x for $(1, x)$ (add the intercept column)
- Conditional log-likelihood of y given x

$$\begin{aligned}\ell(\beta) &= \sum_{i=1}^{n}\log P(Y = y_i|X = x_i; \beta) \\ &= \sum_{i=1}^{n}[y_i \log P(c_1|x_i;\beta) + (1 - y_i)\log P(c_2|x_i;\beta)] \\ &= \sum_{i=1}^{n}[y_i(\beta^\top x_i) - \log(1 + e^{\beta^\top x_i})]\end{aligned}$$

- Unlike with linear regression, there is no closed form solution for $\hat{\beta}$
- Score equation: take derivative with respect to β and set to 0

$$\frac{\partial\ell(\beta)}{\partial\beta} = \sum_{i=1}^{n} x_i(y_i - p(x_i;\beta)) = 0$$

where $p(x_i;\beta) = e^{\beta^\top x_i}/(1 + e^{\beta^\top x_i})$.
- There are $(p + 1)$ equations, nonlinear in β – has to be solved numerically

- Solve with Newton-Raphson algorithm, a standard iterative general numerical optimization algorithm
- For logistic regression, reduces to iteratively reweighted least squares, at each iteration solving a weighted least squares problem of the form

$$\beta^{\text{new}} = \arg\min_{\beta}(z - X\beta)^\top W(z - X\beta)$$

- Requires an initial value β^0; can use $\beta^0 = 0$
- Must be iterated until β does not change anymore; does not always converge.
- If the model is correct, $\hat{\beta}$ is consistent, that is, $\hat{\beta} \to \beta$ as the sample size n grows.
- The distribution of $\hat{\beta}$ converges to $N(\beta, (X^\top WX)^{-1})$.
- Thus $\hat{\beta}$ is asymptotically unbiased ($E\hat{\beta} \to \beta$) as $n \to \infty$.

- For LDA, can also calculate the logit of class odds for classes k and K:

the dimension of xT:1*p
the dimension of ∑:p*p
the dimension of mu: P*1

$$\begin{aligned}\log\frac{P(Y = c_k|x)}{P(Y = c_K|x)} &= x^\top \Sigma^{-1}(\mu_k - \mu_K) + \log\frac{\pi_k}{\pi_K} \\ &\quad - \frac{1}{2}(\mu_k + \mu_K)^\top \Sigma^{-1}(\mu_k - \mu_K) \\ &= \alpha_{k0} + \alpha_k^\top x\end{aligned}$$

=αk0+αk1x1+...αkpxp

- Logistic model:

$$\log\frac{P(Y = c_k|x)}{P(Y = c_K|x)} = \beta_{k0} + \beta_k^\top x$$

- LDA: linearity is a consequence of the Gaussian assumption for the class densities and the assumption of a common covariance matrix.
- For logistic regression, linearity is there by construction.
- The coefficients are estimated differently.
- The joint density of (x, y) is   common component: linear log-odds

$$P(X = x, Y = c_k) = P(X = x)P(Y = c_k|X = x) = p(x)P(Y = c_k|X = x)$$

where $p(x)$ is the marginal density of the input x.
- For both LDA and logistic regression, the term $P(Y = c_k|x)$ has the same logit linear form

$$P(Y = c_k|x) = \frac{\exp(\beta_{k0} + \beta_k^\top x)}{1 + \sum_{k'=1}^{K}\exp(\beta_{k'0} + \beta_{k'}^\top x)}$$

LDA and logistic regression make different assumptions about $p(x)$.
  - The logistic model leaves the marginal density of x arbitrary and unspecified.
  - The LDA model assumes a Gaussian mixture density

$$p(x) = \sum_{k=1}^{K}\pi_k \cdot \phi(x; \mu_k, \Sigma)$$

  - Logistic regression makes fewer assumptions about the data, and is more general.
- LDA is easier to compute than logistic regression.
- If the true $f_k(x)$'s are Gaussian, LDA is better: logistic regression may lose up to 30% efficiency in error rate (Efron 1975).
- LDA uses all the data points to estimate the covariance matrix – more information but not robust against outliers.
- Logistic regression, through interatively reweighted least squares, down-weighs points far from the decision boundary; more robust.
- KNN takes a different approach: completely non-parametric
- No assumptions are made about the shape of the decision boundary
- Main advantage of KNN: deals well with non-linear and highly complex boundaries.
- Main disadvantage of KNN: no inference (no coefficients for the predictors or p-values).   KNN vs LDA/Logistic
- We expect KNN to dominate both LDA and logistic regression when the decision boundary is highly non-linear.
- QDA is a compromise between the completely non-parametric KNN method and the linear LDA and logistic regression.
- The boundary is non-linear, but still of a specified form (quadratic)
- Also makes the normal assumption
- LIkely the best choice when the true decision boundary is:
  - Linear: LDA and logistic regression
  - Moderately non-linear: QDA
  - More complicated: KNN
- When the relationship between Y and X is linear and the number of observations n is much bigger than the number of predictors p ($n \gg p$), the OLS work well (low bias, low variance, highly interpretable)   prediction accuracy

to let OLS work well: linear
independent error
similar variance

- When $n \approx p$, then the least squares fit can have high variance and may result in overfitting and poor predictions.
- When $n < p$, the OLS estimates are not unique and their variance is infinite.
- The multi-collinearity problem: unavoidable when $n < p$
- Advantage: end up with a small linear model
- Disadvantages:
  - Finding the best subset is a combinatorial problem (2^p possible models); greedy search algorithms offer no guarantee of actually finding "the best"
  - Inference may not be valid if you tried many models before choosing "the best".
- Shrinking the estimated coefficients towards zero (in absolute value)
- Shrinkage reduces variance
- If some of the coefficients are shrunk to exactly zero, those variables can be removed.   increase bias
- E.g. ridge regression and the lasso
- Advantages: efficient optimization methods exist; valid inference is being developed
- Disadvanatges: shrinkage increases bias; does not always result in variable selection.
- The RSS will always decrease (and R^2 increase) as the number of variables increases
- The red line tracks the best model for a given number of predictors, according to RSS and R^2
- Consider a candidate model with d predictors, $d \leq p$
- Akaike information criterion (AIC)

σ_p^2=1/(n-p)*RSS_p

$$\text{AIC} = \frac{1}{n\hat{\sigma}^2}(\text{RSS} + 2d\hat{\sigma}^2)$$

where $\hat{\sigma}^2$ is estimated from the model with all p predictors.
- Bayesian information criterion (BIC)

$$\text{BIC} = \frac{1}{n\hat{\sigma}^2}(\text{RSS} + \ln(n)d\hat{\sigma}^2)$$

- Find a model to minimize AIC or BIC, over all values of d and all subsets of variables

- Definition:

$$C_p = \frac{1}{n}(\text{RSS} + 2d\hat{\sigma}^2)$$

- For linear regression, C_p and AIC are equivalent
- In general, they are not the same; C_p aims to estimate the test MSE of the candidate model

$$\frac{1}{n}\sum_i E(\hat{y}_i - Ey_i)^2$$

- Definition of adjusted R^2:

$$\begin{aligned} R_a^2 &= 1 - \frac{\text{RSS}/(n-(d+1))}{\text{TSS}/(n-1)} \\ &= 1 - \left(\frac{n-1}{n-(d+1)}\right)(1-R^2) \end{aligned}$$

- Adding a predictor will only increase R_a^2 only if it has predictive value.
- Ridge regression estimates β's by minimizing

$$\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij})^2 + \lambda\sum_{j=1}^{p}\beta_j^2 = \text{RSS} + \lambda\sum_{j=1}^{p}\beta_j^2$$

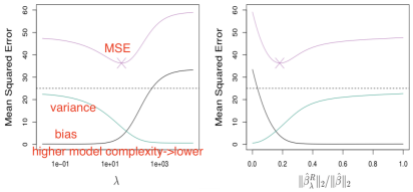
- $\lambda > 0$ is a tuning parameter to be determined is mathematically equivalent to solving the constrained optimization problem

$$\min_{\beta}\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij})^2 \text{ subject to } \sum_{j=1}^{p}\beta_j^2 \le s$$

- s and λ can be mapped to each other (one-to-one correspondence)
- The ℓ_2 norm of a vector β is defined as

$$\|\beta\|_2 = \sqrt{\sum_{j=1}^{p}\beta_j^2}$$

- Same as the usual vector norm in Euclidean space
- What shape does $\sum_{j=1}^{p}\beta_j^2 < s$ define?
- t-statistics and p-values do not change with rescaling in OLS
- The ridge penalty term makes scaling much more important; we need different coefficients to be on the "same footing"
- Thus always standardize predictors before applying a shrinkage method



computational advantage of ridge regression

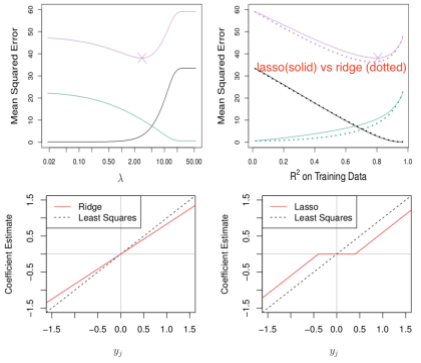
- If p is large, then using the best subset selection approach requires searching through exponentially many possible models.
- Ridge regression is a quadratic optimization problem and can be solved in closed form
- For any given λ, there is a closed form solution
- Ridge regression works when $p > n$ and when predictors are collinear, both situations where OLS fails
- Ridge regression is not perfect: the final model still includes all variables (no selection means harder to interpret)
- The Lasso estimates the β's by minimizing

$$\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij})^2 + \lambda\sum_{j=1}^{p}|\beta_j| = \text{RSS} + \lambda\sum_{j=1}^{p}|\beta_j|$$

- The ℓ_1 norm of a vector β is defined as

$$\|\beta\|_1 = \sum_{j=1}^{p}|\beta_j|$$

- Sometimes referred to as the Manhattan distance
- What shape does $\sum_{j=1}^{p}|\beta_j| \le s$ define?
- Using this penalty, it could be proven mathematically that some coefficients will be set to exactly zero.
- Lasso can produce a model with good predictive power that is still simple to interpret.



- Larger λ means more coefficients set to 0; but frequently better prediction is achieved with a smaller λ
- Sometimes a "one standard error" rule is used to select a more interpretable model: pick the largest λ such that the corresponding CV error is within one standard error (over cross-validation folds) of the best error
- Relaxed lasso: choose predictors with a larger λ, then refit the model with just those predictors without shrinkage (or little shrinkage)
- Tools that involve repeatedly drawing samples from a training set and refitting a model of interest on each sample in order to obtain more information about the fitted model
  - Uncertainty estimation: estimate how much results vary from sample to sample bootstrap
  - Model assessment: estimate test error rates CV
  - Model selection: select the appropriate level of model flexibility CV
- Advantages: the validation set approach
  - Simple
  - Easy to implement
- Disadvantages:
  - The validation MSE can vary a lot from split to split
  - Only a subset of observations are used to fit the model (training data). Statistical methods tend to perform worse when trained on fewer observations.
- LOOCV has less bias: We fit the model using training data that contains $n-1$ observations, i.e. almost all the data, and we do it n times; each point gets to "participate" in the training.
- No randomness: LOOCV will produce the same answer every time because every point is left out in turn, whereas the validation set approach depends on the random split.
- LOOCV is computationally intensive: We fit each model n times!