Notation:

X is a $n \times p$ matrix

$$X = \underbrace{\begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & & & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{bmatrix}}_{p} \Bigg\} n$$

x_{ij} is the (i, j) th element

$i = 1, \ldots, n$ indexes samples / observations

$j = 1, \ldots, p$ indexes variables.

$$x_i = \begin{bmatrix} x_{i1} \\ x_{i2} \\ \vdots \\ x_{ip} \end{bmatrix}$$

row vector of x_i of length p contains p variables / measurements

$$\vec{x_j} = \begin{bmatrix} x_{1j} \\ \vdots \\ x_{nj} \end{bmatrix}$$

Can write as:

$$X = [\vec{x}_1 \ \vec{x}_2 \cdots \vec{x}_p] \quad \text{or}$$

$$X = \begin{bmatrix} x_1^T \\ \vdots \\ x_n^T \end{bmatrix} \quad \text{where} \quad x_i^T = (x_{i1} \ x_{i2} \cdots x_{ip})$$

y_i is the ith observation of the variable on which we make predictions

$$\vec{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}$$

x_i vector of length p

Observed data: $\{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$

Input variable: denoted as X_1, where X_1 might be TV budget, X_2 radio budget etc...

- Inputs also called predictors, independent variables, features

Output variable: denoted as Y.

- Outputs also called the **response**, **dependent variable**

  e.g. sales.

- $X_p = (X_1, X_2, \ldots, X_p)$ p predictors

- $Y = f(X) + \epsilon$
  - f: fixed but unknown function of $X = (X_1, \ldots, X_p)$
  - ϵ: random error, independent of X, w/ $E(\epsilon) = 0$

Prediction:

$\hat{Y} = \hat{f}(X)$ ($\hat{f}$ is a black-box)

- $\hat{f}$ is estimate of f w/ $\hat{Y}$ is resulting prediction of Y
- $\hat{f}$ depends on **reducible error** & **irreducible error**
- ϵ cannot be predicted using X — **irreducible error**
- Average or expected value of be actual and predicted

$$E(Y - \hat{Y})^2 = E[f(X) + \epsilon - \hat{f}(X)]^2$$

$$= E[f(X)^2 - 2f(X)\hat{f}(X) + \hat{f}(X)^2 + \cdots$$

$$\cdots + \epsilon f(X) + \epsilon \hat{f}(X) + \epsilon^2]$$

$$= E[f(X)^2 - 2f(X)\hat{f}(X) + \hat{f}(X)^2 + \epsilon^2]$$

$$= \underbrace{[f(X) - \hat{f}(X)]^2}_{\text{reducible}} + \underbrace{\text{Var}(\epsilon)}_{\text{irreducible}}$$

where $\text{Var}(\epsilon) = \epsilon^2$

- irreducible error provides upper bound on accuracy of estimate of prediction for Y.

Inference :

- Want to understand association between Y and $X_1, \ldots, X_p$.
- Estimate f not necessarily to make predictions
- $\hat{f}$ not black-box

Prediction vs. Inference:

- How much extra will house be worth if it has a view of river? } Inference
- Given this house characteristics, is it overvalued or undervalued? Predict the value given these characteristic } Prediction

How To Estimate f ?

- Training data (observations) used to train our method to estimate f where:
  - x_{ij} is the jth predictor as input for observation i
  - y_i is the ith observation for the response variable
  - $\{(x_1, y_1), \ldots, (x_n, y_n)\}$ is the training data where $x_i = \begin{bmatrix} x_{i1}^T \\ \vdots \\ x_{ip}^T \end{bmatrix}$
- Want $Y \approx \hat{f}(X)$ for observation (X, Y)

Parametric Methods:

1. Assume functional form. If f is linear in X:

$$f(X) = \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p$$

- Linear model assumption instead of estimating p-dim function, $f(x)$, estimate $p+1$ coefficients $\beta_0, \beta_1, \ldots, \beta_p$.

2. After model selection train (fit) model using training data.

For lin. model assumption estimate $\beta_0, \beta_1, \ldots, \beta_p$ s.t.

$$Y \approx \beta_0 + \beta_1 X_1 + \cdots + \beta_p X_p.$$

One approach: least squares.

- Reduces estimating f to estimating a set of parameters
- Flexibility: higher means can fit many functions, more parameters

- **Overfitting**: a more complex model can lead to overfitting the data, the model follows the errors or noise.
- Overfitted model will not yield accurate estimates on new observations (test set)

Non-parametric Models:

- No explicit assumption about functional form of f.
- Can't reduce estimate of f to estimating parameters, instead need very large number of observations to est. f (disadvantage)

Why choose more restrictive model?:

- Restrictive models are more interpretable versus flexible models.
- If goal is inference want a model that's more interpretable
- If goal is prediction don't care about interpretability i.e. predicting stock prices want algorithm that accurately predicts price

Supervised vs. Unsupervised:

- **Supervised learning**: for each observation of predictor measurement x_i, $i = 1, \ldots, n$ there is associated response y_i.
  - Goal of fitting model is to accurately predict the response for future observations (predictions) or better understand relationship between response & predictors (inference)
- **Unsupervised learning**: each observation $i = 1, \ldots n$ observe vector of measurements x_i but no associated response y_i
  - One tool is to use cluster analysis. If there are p variables in our dataset then $p(p-1)/2$ distinct scatterplots. Need automated methods.
- **Semi-supervised learning**: for n obs., $m < n$ have responses, $n-m$ don't.

Regression vs. Classification:

- Quantitative (Numerical) vs. Qualitative (Categorical)
- Regression problems usually quantitative
- Classification problems usually qualitative.
- Choose learning method on basis of whether response is quantitative or qualitative.
- Model accuracy: no free lunch in stats as no one method dominates all others over all possible data sets.
- Mean Squared Error: in regression to eval. performance use MSE

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{f}(x_i))^2$$

where $\hat{f}(x_i)$ is prediction for the ith observation.

- Given training observations $\{(x_1, y_1), \ldots, (x_n, y_n)\}$ obtain estimate $\hat{f}$ then compute $\hat{f}(x_1), \ldots, \hat{f}(x_n)$ and if approx close to $y_1, \ldots, y_n$ then MSE small.

- Unseen test observation more interested in $\hat{f}(x_0)$ equal to y_0, observation not used to train stat. learn. method.
- Average squared prediction error for test observation:

$$\text{Ave}(y_0 - \hat{f}(x_0))^2$$

- Cross-validation: finds min. pt of test MSE vs. flexibility (i.e. degrees of freedom for splines).

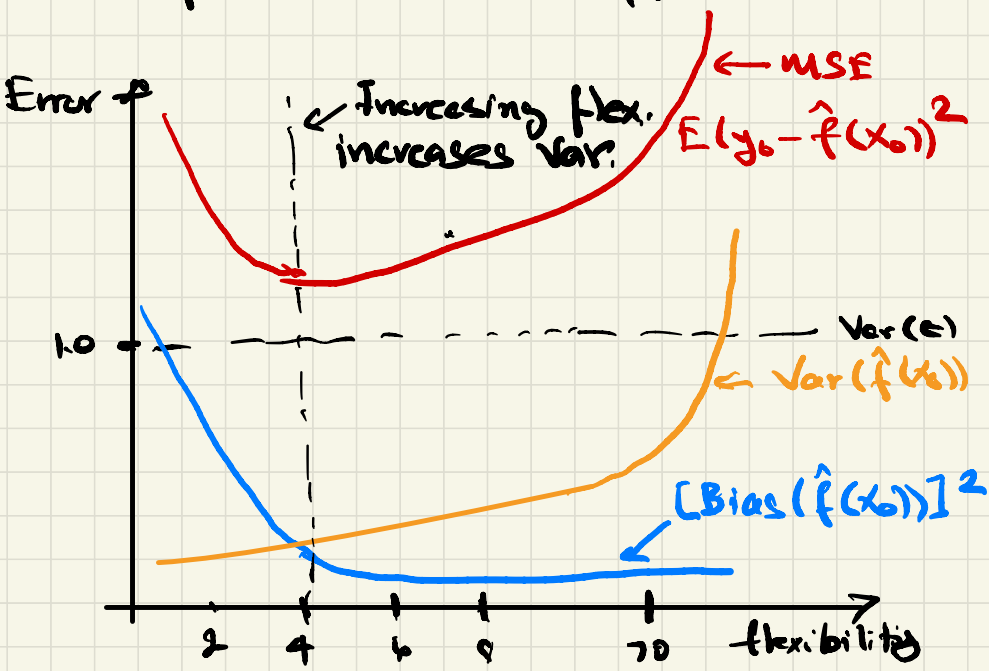
Bias-Variance Trade-off:

- Expected test MSE for x_0 decomposed:

$$E(y_0 - \hat{f}(x_0))^2 = \text{Var}(\hat{f}(x_0)) + [\text{Bias}(\hat{f}(x_0))]^2 + \text{Var}(\epsilon)$$

- Same as avg. test MSE after repeatedly estimating f using large number of training sets and test each on x_0.
- Want learning method w/ low variance & bias.

- **Variance**: amount $\hat{f}$ changes if we estimate using different training set. High variance method means small change in training data can result in large changes in $\hat{f}$.
- More flexible models have higher variance
- **Bias**: error due to approx. complicated situation w/ simple model. i.e. high bias results from lin. reg. to approx. non-linear situation
- More flexible models have lower bias



Bias-variance tradeoff: relationship between bias, variance, and test MSE.

- If f linear, then linear regression will have no bias. More flexible method can't compete. If f non-linear and have large number of training obs then flex. approach possibly better.

Classification: y_i no longer quantitative.

- accuracy of $\hat{f}$ based on training error rate, number of misclassifications for estimate $\hat{f}$

$$\frac{1}{n}\sum_{i=1}^{n} I(y_i \neq \hat{y}_i)$$

- where $\hat{y}_i$ predicted class label for ith observation using $\hat{f}$
- $I(y_i \neq \hat{y})$ is indicator variable it $y_i \neq \hat{y}$ is 1 else 0.

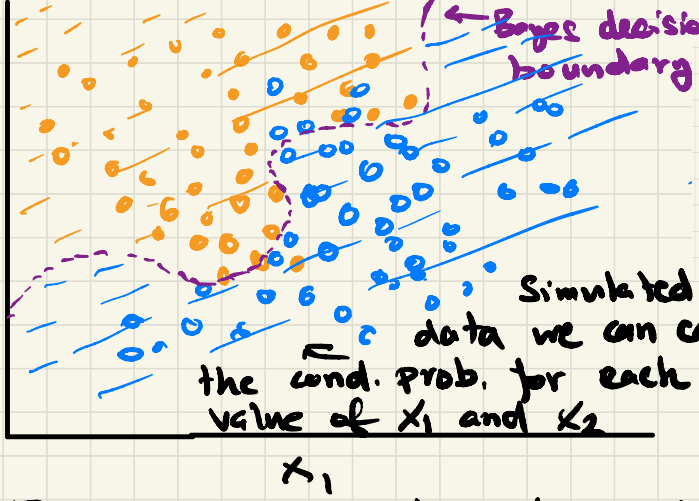
- If $I(y_i \neq \hat{y}_i) = 0$ then ith obs. classified correctly by classification
- Error rate $\frac{1}{n}\sum_{i=1}^{n} I(y_i \neq \hat{y}_i)$ computes fraction of misclassifications.
- Test error rate: test obs. (x_0, y_0)

$$\boxed{\text{Ave}(I(y_0 \neq \hat{y}_0))}$$

- The Bayes Classifier: test error minimized by simple classifier that assigns obs. to most likely class
- Assigns test obs. predictor vector x_0 to class j s.t.

$$\boxed{\Pr(Y = j \mid X = x_0)}$$ conditional prob.

- is the prob. $Y = j$ given observed predictor vector x_0.
- Bayes classifier: 2 class; predicts class 1 if $P(Y=1|X=x_0) > 0.5$ and class 2 otherwise.



- $Pr(Y = \text{orange} \mid X) > 0.5$ points
- $Pr(Y = \text{blue} \mid X) < 0.5$ points
- $Pr(Y = \text{purple} \mid X) = 0.5$ called the Bayes decision boundary

- **Bayes error rate**: lowest possible test error rate produced by Bayes classifier.
  - Bayes classifier chooses class where $Pr(Y = j \mid X = x_0)$ is largest so error rate is $1 - \max_j Pr(Y = j \mid X = x_0)$
  - **Overall Bayes error rate**:

$$1 - E\left(\max_j Pr(Y = j \mid X)\right)$$

- Expectation avgs. prob. over all possible values of X.
- For prev. example Bayes error rate is 0.133. Greater than 0 bc overlap in population so $Pr(Y = j \mid X = x_0) < 1$ for some values of x_0.
- Bayes error rate analogous to irreducible error.

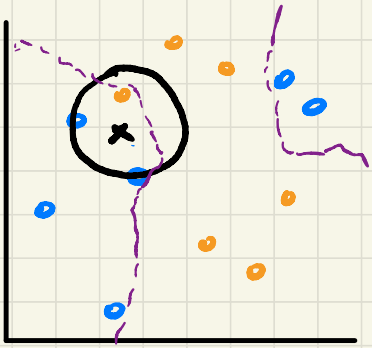
K-Nearest Neighbors:

- For real data we don't know cond. dist. of Y given X so Bayes classifier is gold standard (unattainable).
- Instead estimate cond. distribution of Y given X then classify the obs. to class w/ highest estimated prob.

KNN: given integer K and test obs. x_0, KNN classifier identifies K points in training data closest to x_0 represented by neighborhood N_0. The estimate cond. prob. for class j as fraction of points in N_0 whose response values equals j

$$\Pr(Y = j \mid X = x_0) = \frac{1}{K} \sum_{i \in N_0} (y_i = j)$$

- KNN classifies test obs. x_0 to class w/ largest probability



- K=3, test observation 'x' with closest points identified 2/3 prob for blue and 1/3 prob for orange so KNN predicts 'x' is blue.

- KNN decision boundary can be close to Bayes classifier.
- Low K (K=1): decision boundary overly flexible and finds patterns that don't correspond to Bayes decision boundary → classifier is low bias but high variance.
- High K (K=100): decision boundary close to linear → low variance but high bias.
- Choose correct amount flexibility crucial for regression & classification settings the bias-variance tradeoff, resulting U-shape test error etc