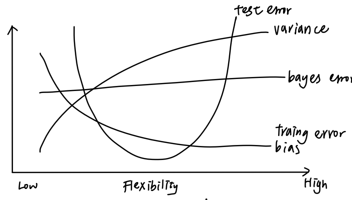
*Basic

$\hat{Y} = \hat{f}(X)$ $\left(\begin{array}{l}\hat{f}: \text{estimate of } f \\ \hat{Y}: \text{resulting prediction}\end{array}\right)$

$E(Y-\hat{Y})^2 = E(f(X)+\varepsilon-\hat{f}(X))^2$
$= \underbrace{(f(X)-\hat{f}(X))^2}_{\text{reductible error}} + \underbrace{Var(\varepsilon)}_{\text{Irreductible error}}$

- accuracy of $\hat{Y}$: reductible error → ↓ by choosing best statistical learning technique
  irreductible error → ε, could not reduced ∵ Y included variable that are not included.



*Accessing model Accuracy

- Regression Problem
  - Mean Square Error (MSE) : measurement of fitting 越小→精確 $MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{f}(x_i))^2$
  - Variance and Bias [Low bias and Low Variance]
    low ① variance : $\hat{f}$ would change when using different dataset (reductible error)
    high ② bias: error introduced by real-world problem (irreductible error)
- Classification Problem
  - Error Rate $= \frac{1}{n}\sum_{i=1}^{n} I(y_i \neq \hat{y}_i)$
  - Bayes Classifier
    ① base on the probability, given each observation the most likely class.
    ② Bayes decision boundary
    ③ Bayes error rate: the lowest possible test error rate (irreductible error) $1 - E(\max_j Pr(Y=j|X))$
  - Logistic Model: $p(X) = \frac{e^{\beta_0+\beta_1 X}}{1+e^{\beta_0+\beta_1 X}} \Leftrightarrow \log\left(\frac{p(X)}{1-p(X)}\right) = \beta_0 + \beta_1 X$ (log odds)
  - K-Nearest Neighbors (KNN) : Unknowing distribution
    ① Choose K (use CV: split data into K groups, and pick 1 as tesing data, choose with min MSE)
    ② Find the points in training data that are closed to X_0
    ③ Calculate the probability $= \frac{1}{K}\sum I(y_i = j)$
    K ↑ → boundaries smoother, linear, loss variance, inflexible

	Flexible	Inflexible
example	decision tree, KNN, SVM	linear, logistic regression
Intepretability	hard	easy
overfitting	easy	hard
complexity	high	low
Advantage	accurate relationship	handle large data
Disadvantage	Inefficient	less accurate
bias	low	high
variance	high (overfitting)	low

	Parametric	Non-parametric
assumption	many	few
Adv./Disa.	easy to Interpret efficient	flexible need large data

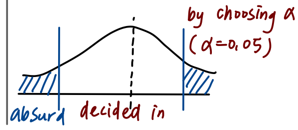
*Simple linear regression

- residual(e): difference between actual and predicted data
- Residual Sum of Squares (RSS) / Total Sum of Squares (TSS) / Estimated Sum of Squares (ESS)
  Fitting linear model : find min RSS
- $aX+bY$, $E(aX+bY) = aE(X)+bE(Y)$, $Var(aX+bY) = a^2Var(X)+b^2Var(Y)$
- Residual Standard Error (RSE) : the measure of lack of fit the model
- Standard Error (SE) $Var(\hat{\mu}) = SE(\hat{\mu})^2 = \frac{\sigma^2}{n}$
- Confidence interval (CI) 95% CI = point estimates ±2SE (68%→1, 99%→3)

$\hat{Y} = \hat{\beta}_0 + \hat{\beta}_1 X$ knowable world
$Y = \beta_0 + \beta_1 X + \varepsilon$ unknowable world
$Y = f(X) + \varepsilon$ ← Irreductible error

$e_i = y_i - \hat{y}_i$
$TSS = \sum_{i=1}^{n}(y_i - \bar{y})^2$
$RSS = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2$
$ESS = \sum_{i=1}^{n}(\hat{y}_i - \bar{y})^2$
$TSS = ESS + RSS$

$R^2 = \frac{ESS}{TSS}$
adding variable → X ↓ R^2 ⇒ overfitting
$RSE = \sqrt{\frac{1}{n-2}RSS}$



△ Outlies correlation : Prof. Rule of Regression
1. Always look at correlation before fitting model.
2. Always plot fitted values v.s. residuals.

*Hypothesis testing $Y = \beta_0 + \beta_1 X$

- H_0: No relationship between X and Y (Null Hypothesis), $\beta_1 = 0$ (t-test)
- H_1: Some relationship between X and Y (Alternative Hypothesis), $\beta_1 \neq 0$
- Type I error (Control by α) : mistakenly reject H_0 while H_0 is True
- Type II error : fail to reject H_0 when H_0 is False
- Multiple Hypo testing : should only see Type I error
  → to identify relationship (i.e. reject H_0)
- F-test
  $H_0: \beta_1 = \beta_2 = \cdots = \beta_k = 0$
  H_A: at least 1 coefficient ≠ 0
- t-statistic : relationship between X and Y
  $t = \frac{\hat{\beta}_1 - 0}{SE(\hat{\beta}_1)}$

- Correlation (COR)
  $Cor(X,Y) = \frac{\sum(X_i-\bar{X})(y_i-\bar{y})}{\sqrt{\sum(X_i-\bar{X})^2}\sqrt{\sum(y_i-\bar{y})^2}}$

① relationship between predictors & response
③ identify variable
② model fit RSE, R^2

*Model Selection

① Best Subset/All subset : Fit all possible model
  Q: 2^p of model, difficult to complete
② Foward Selection (從1→p個variable) ③ Backward Selection (從p→1個variable)
④ Foward Stepwise: Correlation between variable.
  null model → add single variable at each step
  & remove variable no longer significant
⑤ False Discovery rate selection : variable entered if lowest p-value & p-value < FDR

$FDR = \frac{q \cdot k}{p}$ where $\begin{cases} p = \text{number of variables} \\ k = \text{number of variable in model } (k \leq p) \\ q = E\left[\frac{\#\text{False Discovery}}{\#\text{Total Discovery}}\right] \end{cases}$ △Reject when True → Type I Error

- Evaluation Critirea
  1. min RSS
  2. F (H_0: all $\beta_i = 0$)
  3. p-value of most recently add variable (lowest)
  4. Adjusted $R^2 = 1 - \frac{RSS/(n-d-1)}{TSS/(n-1)}$
  5. $C_p: \frac{1}{n}(RSS + 2d\hat{\sigma}^2)$
  6. Bayes Information Criteria (BIC) $= \frac{1}{n}(RSS + \log(n)\cdot d \cdot \hat{\sigma}^2)$
  7. Aikake Information Criteria (AIC) $= \frac{1}{n\hat{\sigma}^2}(RSS + 2d\hat{\sigma}^2) = \frac{C_p}{\hat{\sigma}^2}$
  8. CV

⑥ Shrinkage/Regularization λ: the extent of shrinkage, use CV to choose (MSE for test v.s. train)
- Ridge regression $\min RSS + \lambda(\sum\beta_j)^2$
  - Advantage: bias-variance tradeoff & computationally efficient
    λ ↑ → flexibility ↓ → variance ↓, bias ↑
    λ = 0 → variance high, no bias
  - Disadvantage: prediction accuracy in large p.
    (will not set any of them to zero.)
- Lasso $\min RSS + \lambda|\sum\beta_j|$ ← better for model selection
  - Advantage: λ ↑ → coefficient ≈ 0
    (Similar to variable selection)
    → create sparse model with left variable

	K=sample size	K=1
Bias, Variance	high, low	low, high
Decision boundary	oversimplified, smooth X capture structure	complex overfitting

- change dramatically with small change in training data
- capture all data

*R-code

- rep(times, each) : repeat values in a vector
- seq(start, end, length, by)
- data.frame(vectors)
- matrix(values, nrow, ncol)
- read.csv()
- sum(is.na()) : check na value
- str(df) : show column name, type, values.
- summary(df) : show statistical data
- dim(df) : numbers of rows & columns
- as.factor() : change type
- level() : check level of factor
- rnorm(size, mean, sd)
- rbind() : combine 2 dataframe
- subset(df, condition)
- pairs(df) : create matrix of scatterplots
- cor(x, y) : calculate correlation
- lm(formula, data) : fit linear model
- plot(model) hist(vector)
- sample(x, size, replace, prob)
- rbinom(x, size, prob) : binomial distribution
- glm(formula, data, family="binomial") : generalized lm
- knn(train, test, cl, k) : KNN classification
- predict(model) ifelse(predictions, Yes, No)
- mean(predictions != actual) calculate error rate

*KNN classifier v.s. KNN Regression

KNN classifier	KNN Regression
categorial, qualitative → class of category 取鄰近K筆data, 將最有可能(機率最高)的做為predict class.	continuous, quantitative → numerical value 取鄰近K筆data, 將他們的y平均做為prediction

*Resampling → repeatedly draw from training data, with replacement, and refit model or reestimate parameter

1. Cross Validation : estimate test error

- Validation set approach
  ① dataset randomly splited into 2 parts : training set (train (same)) & validation set (predict (new))
  ② Error rate for validation set : MSE (mean squared error)
  ③ repeat several times with different random split
  ④ Result : select model & assess accuracy.
- Leave-one-out Cross-validation (LOOCV)
  ① Split data [training set : only 1 observation (X_1, y_1) ; validation set : rest of n-1 observation $[(X_2,y_2)\cdots(X_n,y_n)]$
  ② Compute $MSE_n = (y_n - \hat{y}_n)^2$
  ③ repeat n times
  ④ Result : $CV_{(n)} = \frac{1}{n}\sum_{n=1}^{i} MSE_i$ / For classification: $CV_{(n)} = \frac{1}{n}\sum_{i=1}^{n} Err_i = \frac{1}{n}\sum_{i=1}^{n} I(y_i \neq \hat{y}_i)$
- K-fold Cross-Validation
  ① split data to k equal size groups [training set : k-1 folds ; validation set : choose 1 fold.
  ② Compute MSE
  ③ repeat k times　△Usually use k=5 or 10
  ④ Result : $CV_{(k)} = \frac{1}{k}\sum_{i=1}^{k} MSE_i$
  → Evaluation [MSE : model performance ; min MSE : choose best model

Feature	LOOCV	k-Fold CV
Number of Folds	Equal to the number of observations in the dataset	Typically a smaller number, such as 5 or 10
Training Set Size	N-1 (where N is the total number of observations)	Approximately (N/k) * (k-1)
Test Set Size	1	Approximately N/k
Computational Cost	High (as the model is trained N times)	Lower (as the model is trained k times, usually k << N)
Variance of Estimate	Low (due to the large number of training runs)	Higher (depends on k, smaller k means lower variance)
Bias	Low (nearly unbiased, as nearly all data is used for training)	Slightly higher (depends on k, larger k means lower bias)
Sensitivity to Data Split	Low (every data point is used for validation exactly once)	Higher (depends on how data is shuffled and split)
Best Use Case	Small datasets due to computational expense	Larger datasets or when computational resources are limited
Generalizability	Can lead to overestimation of performance (due to high similarity between training and test sets)	Better generalizability (more variation in training sets)

2. Bootstrap 自助抽样法/自助重抽法

→ when data is not normal distribution, sampling by randomly choose n of data (with replacement)
* key application : estimating sampling distribution by each mean. , SE = hypo tests, CI's

- Linear Regression　$SE = \sqrt{\frac{\lambda}{n}}$
  ① Collect data
  ② Use assumption of Poission Distribution
  ③ Use Bootstrap : Repeatedly stimulate
- Logistic Regression (Classification)
  ① Collect Data
  ② Use $p = \frac{1}{n}$
  ③ Use Bootstrap : Repeatedly stimulate
- The parametric bootstrap
  1. Fit a parametric model
     → $\hat{f}(x)$ to $X_1, \dots, X_n$ using a parameter estimate $\hat{\theta}$ for θ
  2. for $i = 1, 2, \dots, \beta$
     a) simulate $X_1^*, \dots, X_n^* \overset{i.i.d.}{\sim} f(x)$
     b) compute the statistic $T^* = T(X_1^*, \dots, X_n^*)$ using data $X_1^*, \dots, X_n^*$
  3. Return empirical standard deviation of T^* across the β simulation
- The nonparametric bootstrap
  1. Supposed we are interested in SE of statistic $T = T(X_1, \dots, X_n)$
  2. for $i = 1, 2, \dots, \beta$
     a) Simulate $X_1^*, \dots, X_n^*$ as n samples with replacement from original data $X_1, \dots, X_n$
     b) Compute the statistic $T^* = T(X_1^*, \dots, X_n^*)$ using data $X_1^*, \dots, X_n^*$
  3. Return the empirical standard deviation of T^* across the β simulation.

	KNN	Logistic	Bayes
Assumptions	Few	linear	independence
complexity	high keep all data	moderate	low
interpretability	low	high	moderate
training efficiency	low	high	high
handle large data	poor (train cost)	good	good
Noise	sensitive	moderate	sensitive
high dimensional	poor	good	vary

non-paramethic　binary classification

*Loss & Empirical Risk

- quatratic loss : $\ell(\hat{y}, y) = (\hat{y} - y)^2$
- absolute loss : $\ell(\hat{y}, y) = |\hat{y} - y|$
- Empirical Risk : $\mathcal{L}(\theta) = \frac{1}{n}\sum_{i=1}^{n} \ell(\hat{y}, y)$
- ERM (minimization) : $\min_{\theta} \mathcal{L}(\theta)$ genetic parameter.
- RERM (regularized) : $\min_{\theta} \mathcal{L}(\theta) + \lambda \cdot r(\theta)$ when $r(\theta) = \theta_1^2 + \cdots + \theta_p^2$ (Ridge)
  $r(\theta) = |\theta_1| + \cdots + |\theta_p|$ (Lasso)
  λ ↑ hyperparameter.

*Indicator Variables :

$I_{\{female=1\}} \equiv \begin{cases} 1 : \text{variable = female} \\ 0 : \text{otherwise} \end{cases}$

- interaction terms : $\beta_1 X_1 X_2$
- transformations : $\beta_1 X_1^2$　$\beta_1 \cdot \ln(X_1)$

△ Potential Problems :

1. non-linear f
2. e_i are correlated
3. σ^2 not constant
4. outliers, leverage points.
5. collinearity between X's

*Decision Rule

- generalized linear methods (glm)
- link function : η "eta" transforms $E(Y|X_1, \dots, X_p)$ so that the transformed mean in a linear function of X's

① $\eta(\mu) = \mu$　(OLS)
② $\eta(\mu) = \log(\frac{\mu}{1-\mu})$　logistic regression