card closing → should we try to retain the customer.

~~[illegible]~~
Credit card → rewards, lounges, interest free credit
↓
merchants dislike it due to transaction fees
↓
user wants so merchants end up in non negotiable position

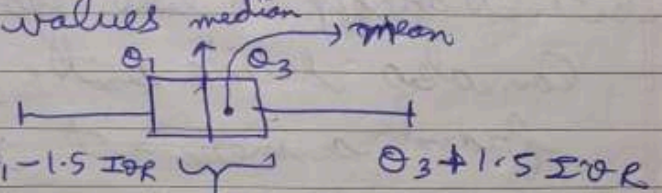
Machine Learning Revision

Missing values → MCAR, not MCAR, Removing rows/cols
fillna ( ) df.dropna(axis=x) ← row/col

Mean median mode imputation → impacts correlations distorts variances

mean → preserves mean of data, numerical
median → robust to outliers
mode → categorical

predicting missing values

Outliers → 

can be a mistake so we remove
Removes 0.7% data from a normal

$$\text{zscore} = \frac{x-\mu}{\sigma} \rightarrow > 3 \text{ ?}$$

↓
How many std deviations away from mean we are ?

Feature Scaling → Suppresses outliers, fastens optimization

Standard Scaling $x' = \frac{x - \mu}{\sigma}$ → makes $\mu = 0$, $\sigma = 1$ → good for gaussian data

Minmax scaling $x' = \frac{x - \min(x)}{\max(x) - \min(x)}$

↓ makes data [0, 1]

Robust Scaling → $x' = \frac{x - Q_1(x)}{Q_3(x) - Q_1(x)}$

↓ Good for removing outliers

Encoding categorical features

OHE → One hot encoding

Ordinal Encoding

Count, frequency encoding

Target guided encoding

Num to cat

Binning

Quantization

Prediction Multi Class

OvA → One vs All

OvO → One v/s One

Cross validation →

Train – validate – test split

k Fold cross validation

Stratified k fold cross validation

LOOCV Leave one out cross validation

68 – 95 – 99.7

Imbalanced Dataset → Underampling

↓ k Means, near Miss, Deletion

Overampling

↓ SMOTE, Adasyn, Duplication

~~Multicollinearity, Dummy variable, Bias-Variance, Power transform, Outliers, Pca, DBSCAN, correlation~~

SMOTE → Synthetic Minority Oversampling 2002

"SMOTE works by selecting examples close in feature space (KNN kinda), drawing line b/w them & drawing new sample along that line"

ADASYN → Adaptive Synthetic Sampling

↳ generate more samples where the density of minority class is less

↳ It ignores majority class hence may have issues at border

Borderline SMOTE → border of classes complified

~~Power transforms~~

Power transforms → log transform, box-cox, yeo johnson

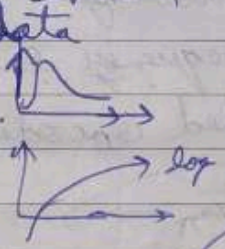
↓ to make data gaussian

↓ WHY?

↓ some ML algo assume data gaussian, error gaussian, makes data smoother hence decision boundaries easier to make

log transform → right skewed data

$\log(x)$



box-cox → $\frac{x^{\lambda} - 1}{\lambda}$

Correlations

↓ Pearson, Spearman, Point Biserial, Phi coefficient

Pearson → $\frac{\sum (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum (x_i - \bar{x})^2} \sqrt{\sum (y_i - \bar{y})^2}} = \frac{\text{Covariance}(X, Y)}{\text{Stdev}(X)\ \text{Stdev}(Y)}$

↓ (σ)

linear correlations

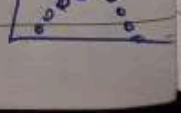
↓

sensitive to outliers

Correlation does not imply causation

~~Correlation~~

Dependency $\not\Rightarrow$ pearson correlation



Spearman correlation → captures monotonicity well
↳ doesn't work for 

$$\rho = 1 - \frac{6 \sum_i d_i^2}{n(n^2-1)}$$ → difference in ranks of variables

rank = sort them & get position / "rank"

If ties then use mean
eg → 5th & 6th rank
rank = 5.5

Multicollinearity
↳ reduces explainability
VIF (Variable Inflation Factor)

$$VIF = \frac{1}{1-R^2}$$

→ fix by dropping correlated columns

$VIF \in [1, \infty)$

R^2 → how much variance of y explained by x

1 → not correlated
1-5 → moderately correlated
VIF > 5 → highly

Dummy Variable trap
Happens in OHE

GLM (Generalized Linear Model)

Linear Regression → $y_i \sim N(\mu_i, \sigma)$; $\mu_i = b_0 + b_1 x_i$

Poisson Regression → $y_i \sim \text{Poisson}(\lambda_i)$; $\ln \lambda_i = b_0 + b_1 x_i$

for Discrete y
count data

distribution fn

link fn

Logistic Regression → $y_i \sim \text{Bern}(p_i)$;

$$p_i = \frac{1}{1 + e^{-(b_0 + b_i x_i)}}$$

Poisson →

$$P(X=x) = \frac{e^{-\lambda} \lambda^x}{x!}$$

$$\ln\left(\frac{p_i}{1-p_i}\right) = b_0 + b_i x_i$$

Bernoulli

$^1C_x \, p^x (1-p)^{1-x}$

n = no of predictor vars

$\text{logit}(p_i)$
↳ logistic link fn

$^1C_x \, p^x (1-p)^{1-x}$

p for x = 1, 1-p for x = 0

x = outcome

Linear Regression → Linearity, Homoscedasticity, Non multicollinearity, Independent observations, Normality of errors

Logistic Regression loss fn = log loss, cross entropy loss

$J(\theta) = -y \log(h_\theta(x)) - (1-y)\log(1-h_\theta(x))$

$h_\theta(x)$ = predicted value $= \frac{1}{1+e^{-\theta^T x}}$ = $P(y=1 \mid x, \theta)$

sigmoid / Logistic fn

Ridge Regression (L2) → $\lambda \sum |a_i|^2$

Lasso Regression (L1) → $\lambda |a_i|$ → can eliminate variables

Elastic Net Regression → $\lambda_1 |a_i| + \lambda_2 |a_i|^2$

OLS Regression → Ordinary Least Squares

Huber loss →  → Huber regression ↓ anti outliers

Multinomial Logistic Regression

$P(\text{Class } 1 \mid X) = \frac{e^{w_1 x + b_1}}{\sum_i e^{w_i x + b_i}}$ → softmax fn

Naive Bayes → Bernoulli NB (predictors are boolean)

Gaussian NB (predictors are continuous, gaussian)

Multinomial NB (predictors discrete, e.g.)

requires independent predictors!

Pseudocount | Laplace smoothing

$P(E_i \mid A) \propto \underbrace{P(A \mid E_i)}_{\text{probality of observing such input given output } E_i} \underbrace{P(E_i)}_{\text{priori probablity}} = \frac{P(A \mid E_i)\, P(E_i)}{P(A)}$

$P(y \mid x_1, x_2 \ldots x_n) \propto P(y) \prod_{i=1}^{n} P(x_i \mid y)$

Precision $= \frac{TP}{TP + FP}$ Recall / Sensitivity / True +ve rate $= \frac{TP}{TP + FN}$

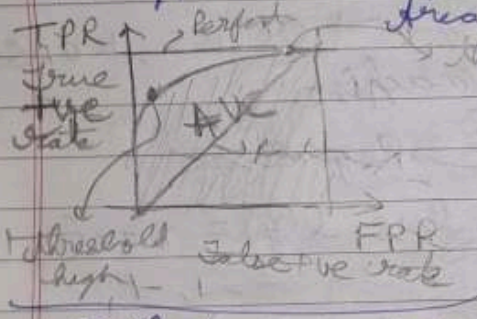
F1 score $= \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$

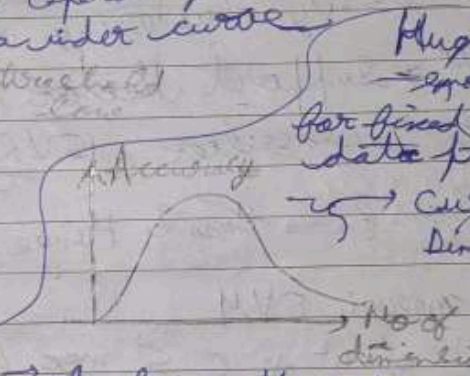
Threshold ↑
precision ↑
recall ↓

Specificity / TNR (True -ve Rate) $= \frac{TN}{TN + FP}$

False +ve Rate $= \frac{FP}{FP + TN}$ = just opposite of Recall

ROC / AUC Receiver Operating Characteristics
Area under curve





Hughes Phenomenon
for fixed no of data pts
→ Curse of Dimensionality

Clustering → dist → Euclidean, Manhattan, Gower, Hamming, Minkowski, Cosine
K-Means
Hierarchical Clustering → Dendograms
Elbow method
Agglomerative, Divisive
Silhouette score = [-1, 1]
$\frac{(b-a)}{\max(a, b)}$
b = avg inter cluster dist
a = avg intra cluster dist
0 → insignificant
1 → perfect

K-mode clustering → categorical data
K-medoids → helps against outliers
K-prototype clustering → (num + cat) data
K-means ++ → Better initializations
Density Based Clustering
DBSCAN → Density Based Spatial Clustering Applications with Noise

~~Hierarchical~~

Hierarchical clustering, SVC, SVM, Hyper parameter, Curse of dim

~~DBSCAN~~

DBSCAN → 2 parameters

→ epsilon → ~~min~~ dist to consider 2 pts as neighbours

minpoints → no of points needed within epsilon dist

→ no need to specify no of clusters

Disadv → some data pts may remain unclustered

parameter sensitive

Cannot handle variable densities

other density clustering algs → OPTICS, HDBSCAN

SVM → Support Vector machine

SVC → classifier SVR → Regressor

SVM loss fn → ~~Tangl~~ Hinge loss $l(y) = \max(0, 1-ty)$

$t = 1, -1$

Soft margin SVM

$$\min_{w} \frac{1}{2} \sum_{i} w_i^2 + C \sum_{i=1}^{m} \max(0, 1 - t_i y_i)$$

$C \downarrow$ → regularization ↑

Hard margin SVM → no regularization

SVM → large margin classifier

Support vectors → data pts closest to decision boundary

Dist b/w hyperplane & support vectors → margin

SVM eqn → $f(x) = \text{sign}(w^T \vec{x} + b)$

Support vector regression (SVR) → maximises pts b/w data boundary

Kernels → Gaussian, polynomial, RBF (Radial Basis fn)

GMM → Gaussian Mixture Model

Soft clustering, Hard clustering

GMM → EM (Expectation Maximization algorithm)

$p(x) = \sum_{i=1}^{k} \phi_i \mathcal{N}(x \mid \mu_i, \sigma_i)$ E step → $\hat{\gamma}_{ik} = \frac{\phi_k \mathcal{N}(x_i \mid \mu_k, \sigma_k)}{\sum_{j=1}^{k} \phi_j \mathcal{N}(x_i \mid \hat{\mu}_j, \hat{\sigma}_j)}$

weight of each datapt ~~gaussian~~ probability i^{th} datapt in k^{th} gaussian

M step →

gaussian
- mean = mean datapts
- weight → avg of probs
- variance → weighted variance of all pts

Kmeans loss fn → SSE sum sq errors

GMM → maximises log likelihood

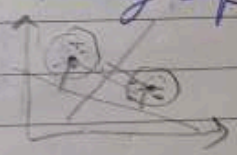
$$\log P(X \mid D) = \sum_{i=1}^{n} \log\left(\sum_{j=1}^{k} \pi_j \mathcal{N}(x_i \mid \mu_i, \sigma_i)\right)$$

PCA (Principal Component Analysis) → eigenvalues, eigenvectors, scree plot, SVD (Singular ~~Vector~~ Value Decomposition)

Dimensionality redn → PCA, LDA, t-SNE

LDA → Linear Discriminant Analysis → assumes gaussian, linearly separable

Fischer LDA → $J(w) = \frac{|\mu_1 - \mu_2|^2}{s_1^2 + s_2^2}$



PCA done using SVD → $X = USV^T$

t-SNE → stochastic Neighbour Embeddings

SNE → 2002

tSNE → 2008

Perplexity → effective no of neighbours for which we calculate similarity

High dim similarity → $p(j \mid i) = \frac{\exp(-\|x_i - x_j\|^2 / 2\sigma_i^2)}{\sum_{k \neq i} \exp(-\|x_i - x_k\|^2 / 2\sigma_i^2)}$

$q(j \mid i) = \frac{(1 + \|y_i - y_j\|^2)^{-1}}{\sum_{k \neq i} (1 + \|y_i - y_k\|^2)^{-1}}$ → low dim similarity

$C = \sum_i KL(P_i \| Q_i) = \sum \sum_i p_{j|i} \log \frac{p_{j|i}}{q_{j|i}}$

KL Divergence → optimise this

Preserves neighbours / local relations

Non deterministic algorithm

Crowding problem avoid

$T \to \frac{\bar{x} - \mu}{s/\sqrt{n}} \sim t_{n-1}$

Ensemble → Bagging
Bootstrap Aggregating

Boosting → Gradient Boosting

→ Adaboost
Adaptive boosting (weights)

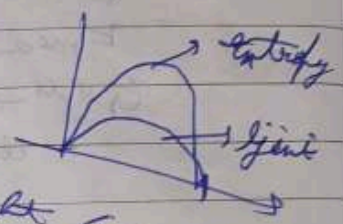
Classifiers
↳ Discriminative
Generative

Decision Tree
CART (Classification & Regression Trees)

Entropy $= \sum_{i=1}^{c} -p_i \log_2(p_i)$

← Information gain → categorical
Gini index → continuous

$1 - \sum_{i=1}^{c} p_i^2$



Cost fn $J(k, t_k) = \frac{m_{left}}{m} G_{left} + \frac{m_{right}}{m} G_{right}$

single feature
threshold

MSE for continuous

m → no of instances

G → impurity

Bagging
Majority voting

Bias Variance Tradeoff