SUPERVISED ALGORITHMS

MACHINE LEARNING

SUPERVISED LEARNING MODEL LIST

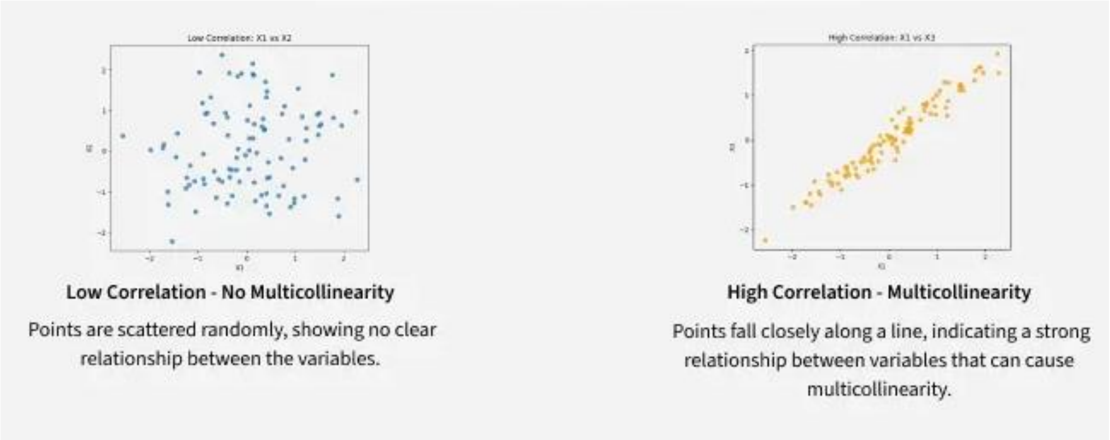
- Linear Models (**linear regression, logistic regression, Polynomial regression**)
- Linear & Quadratic Discriminant Analysis (LDA / QDA)
- Kernel Ridge Regression (KRR)
- **Support Vector Machines (SVM)**
- Stochastic Gradient Descent (SGD)
- Nearest Neighbors (**KNN**)
- Gaussian Processes (GP)
- Cross Decomposition
- **Naive Bayes**

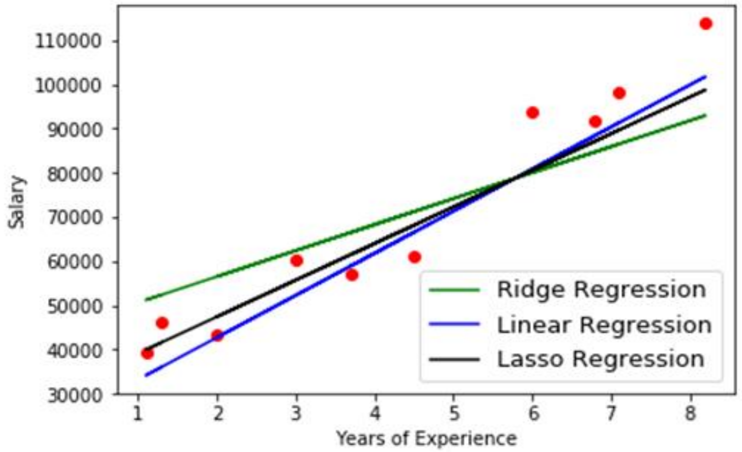
- **Decision Trees**
- **Ensemble Methods**
- Multiclass & Multi-output Learning
- Feature Selection
- Semi-Supervised Learning
- Isotonic Regression
- Probability Calibration
- Neural Networks (Supervised)

SUPERVISED LEARNING

- Ridge Regression is a regularization technique for linear models that prevents **overfitting** and handles **multicollinearity** by adding a penalty (L2 norm) or L2 penalty to the cost function, shrinking coefficients towards zero but never completely eliminating them, thus improving model stability and generalization to new data
- Supports both regression and classification problems.

When to Use It:
•When you have many features and correlated (multicollinearity).
•To prevent unstable coefficient estimates and reduce model variance.
•To build more robust models that perform well on new data (better generalization)





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

Ridge regression is very similar to the method of least squares, with the **exception that the coefficients are estimated by minimizing a slightly different quantity**.

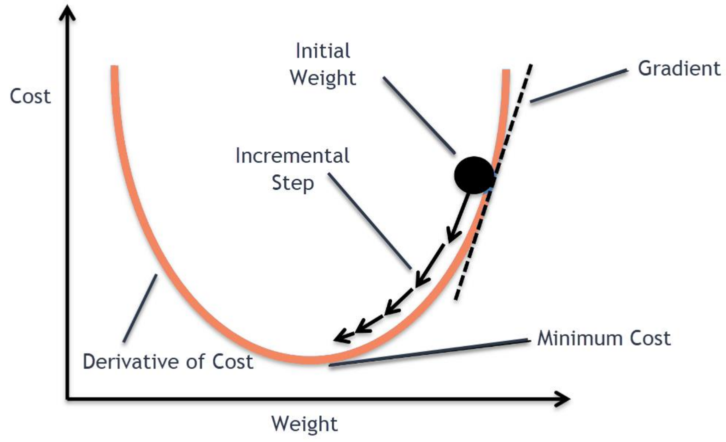
In reality, it's the same quantity, just with something more, with something we call a **shrinkage penalty**.

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

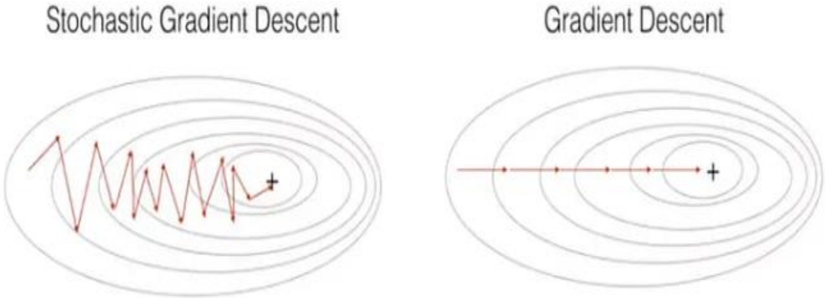
https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.Ridge.html#sklearn.linear_model.Ridge

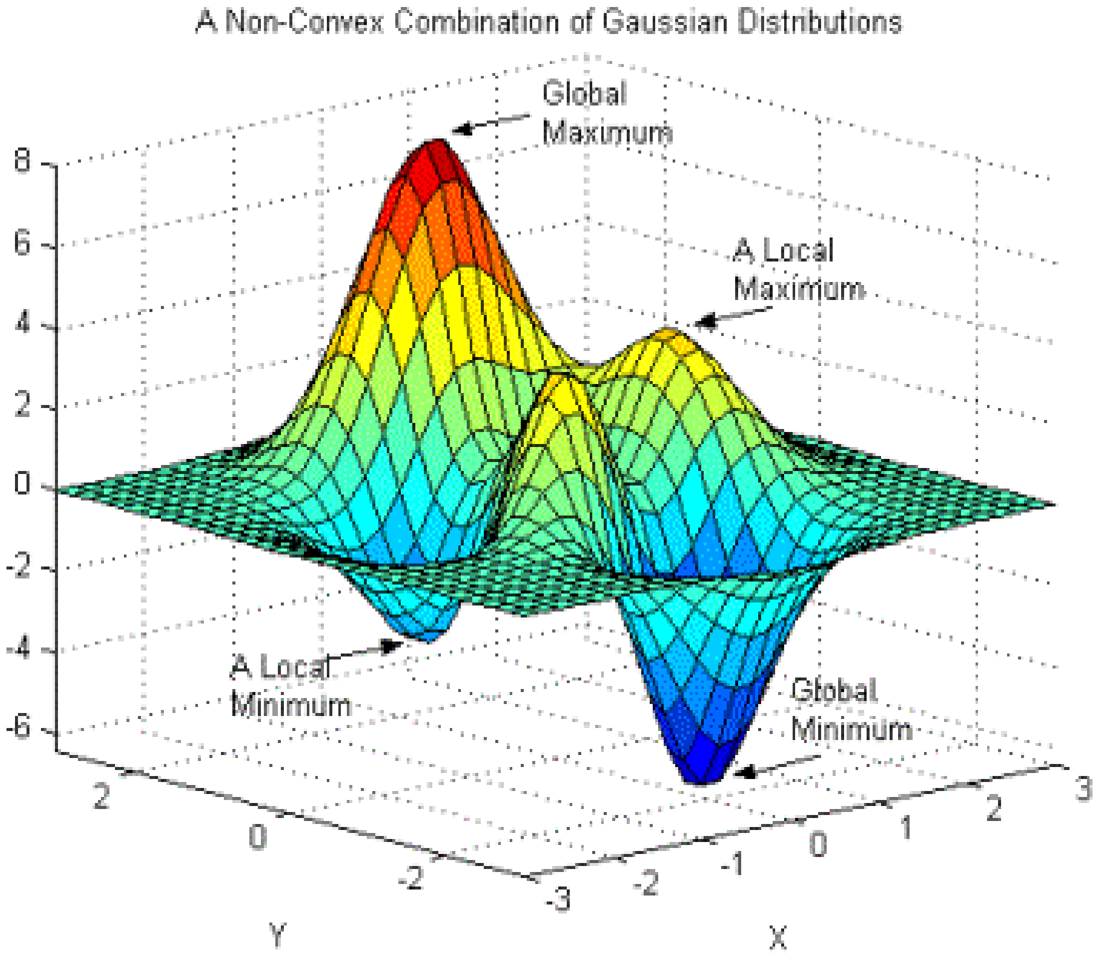
STOCHASTIC GRADIENT DESCENT (SGD)

- **optimization algorithm** widely used in machine learning and deep learning to train models efficiently, especially on large datasets
- The primary goal of SGD is to minimize a **loss function**, which measures the error of a model's predictions.



- Pros: efficiency, scalability, online learning (real-time)
- Cons: Noisy update, need sensitive hyper parameter tuning (learning rate)





SGD **supports both regression and classification problems**

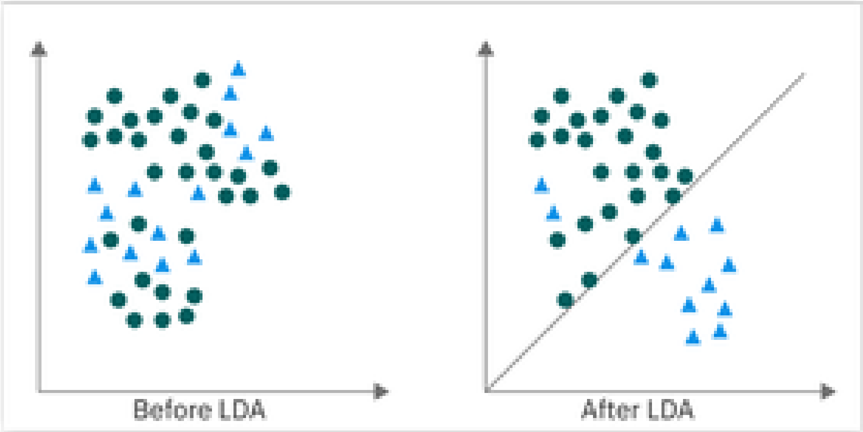
How it Works:

1) Initialize **weight and bias for the model**
2) Algorithm iterates through **training samples (mini-batch)**
3) **Gradient calculation** : algorithm calculates the gradient of the loss function
4) **Parameter update:** model parameters are updated in the opposite direction of the gradient to reduce the loss
5) **Convergence**: local or global minimum.

https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.SGDClassifier.html#sklearn.linear_model.SGDClassifier
https://scikit-learn.org/stable/modules/sgd.html

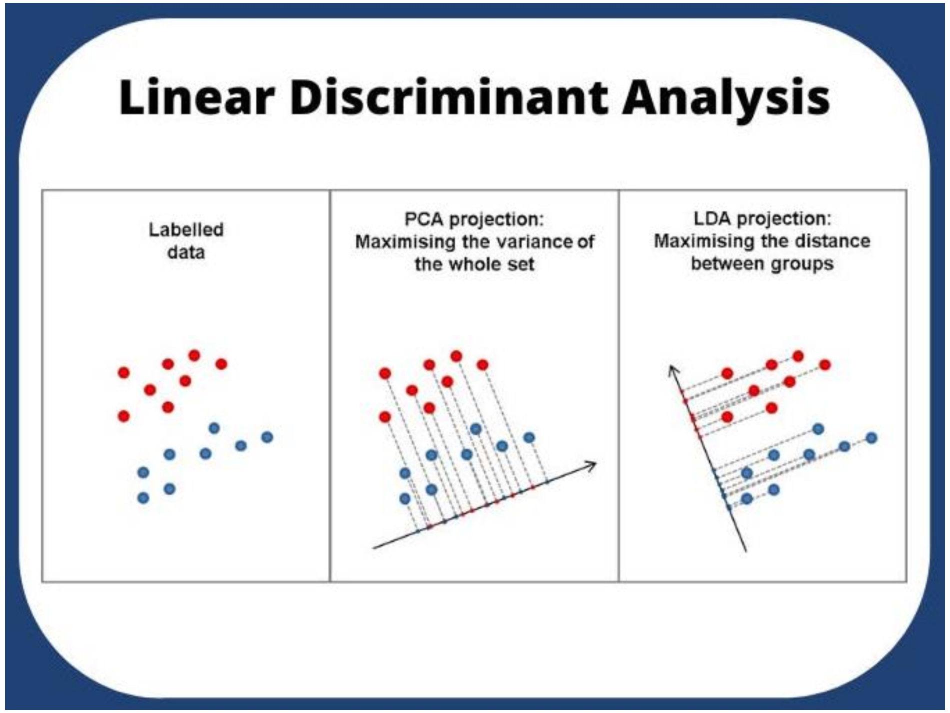
LINEAR DISCRIMINANT ANALYSIS (LDA)

- Linear Discriminant Analysis (LDA) is a powerful supervised machine learning technique used for both **classification** and **dimensionality reduction**
- ***finding the best linear combination of features to separate multiple classes*** by maximizing the ratio of **between-class variance to within-class variance**, creating distinct, lower-dimensional projections for better model performance
- It projects **high-dimensional data onto a new**, **lower-dimensional subspace** where classes are as spread out as possible while staying compact within themselves
- Key Uses: Classification, Dimensionality Reduction, Feature Extraction



How LDA Works

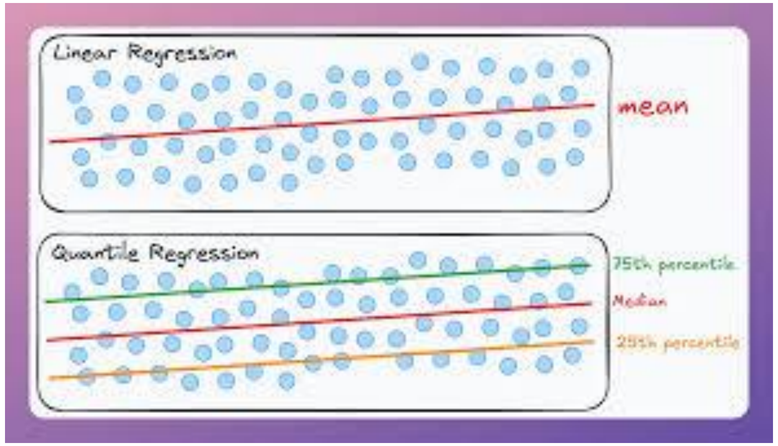
1. **Find Class Means & Covariances**: Calculates the average (mean vector) and spread (covariance) for each class in the original feature space.
2. **Compute Scatter Matrices**: Determines the "within-class" scatter (how spread out points are within a class) and "between-class" scatter (how far apart class centers are).
3. **Find Optimal Projection**: Solves for eigenvectors/eigenvalues to find a new axis (discriminant function) that maximizes the ratio of between-class variance to within-class variance.
4. **Project Data**: Transforms the original data onto this new axis, creating a new, lower-dimensional dataset that emphasizes class separation.

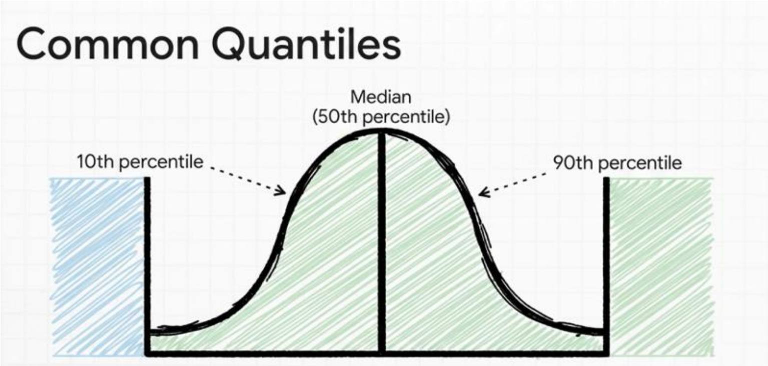


https://scikit-learn.org/stable/modules/generated/sklearn.discriminant_analysis.LinearDiscriminantAnalysis.html#sklearn.discriminant_analysis.LinearDiscriminantAnalysis

QUANTILE REGRESSION

- **Powerful statistical method**
- Use specific **quantiles (like the median, 25th percentile, or 90th percentile)** [quantile can be modified with help of quantile parameter according to our use case] of a response variable's distribution instead of traditional linear regression (mean)
- **When to use** - while analyzing skewed data, handling outliers, and understanding heterogeneous effects
- **Heterogeneous Effects -** predictor changes across different parts of the outcome distribution (e.g., income affects low earners differently than high earners).
- **Loss Function -** Instead of minimizing squared errors. it minimizes a weighted sum of absolute errors





https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.QuantileRegressor.html#sklearn.linear_model.QuantileRegressor

THANK YOU!