| **Feature** | **Supervised Learning** | **Unsupervised Learning** | **Transfer Learning** | **Reinforcement Learning** | **Self-Supervised Learning** | **Semi-Supervised Learning** |
| --- | --- | --- | --- | --- | --- | --- |
| **Definition** | Learns from labeled data to make predictions | Learns from unlabeled data to find patterns | Transfers knowledge from one task to another | Learns through trial & error, using rewards & penalties | Uses unlabeled data but defines structure internally | Combines labeled & unlabeled data for learning |
| **Data Requirements** | Requires large labeled datasets | Works with unlabeled data | Uses pre-trained models on new data | Requires an environment with rewards | Can self-generate pseudo-labels | Needs some labeled and mostly unlabeled data |
| **Decision Boundary** | Typically linear or non-linear | Clustering, density estimation | Adapts from a pre-trained model | Learns optimal policies for actions | Can be complex depending on task | Similar to supervised learning but adapted |
| **Estimation Approach** | Direct function approximation (mapping inputs to outputs) | Pattern recognition, clustering, dimensionality reduction | Fine-tuning a model from previous tasks | Policy optimization (Markov Decision Processes) | Self-labeling, contrastive learning | Partial supervision using limited labels |
| **Robustness** | Effective with well-labeled data | Can discover hidden patterns without labels | Handles new tasks well but depends on source model | Adapts to new environments dynamically | Strong generalization properties | Good balance between labeled and unlabeled data |
| **Best Used For** | Classification, regression, NLP, image recognition | Clustering, anomaly detection, representation learning | Using existing models for new problems | Robotics, gaming, autonomous systems | Large-scale AI models, deep learning architectures | When labeling data is expensive but unlabeled data is available |

**Machine Learning Types and Methods**

| **Learning Type** | **Definition** | **Common Methods** | **Pros** | **Cons** | **Best Used For** |
| --- | --- | --- | --- | --- | --- |
| **Supervised Learning** | Learns from labeled data to make predictions | - Linear Regression<br>- Logistic Regression<br>- Decision Trees<br>- Random Forest<br>- Support Vector Machines (SVM)<br>- Neural Networks | High accuracy with labeled data, interpretable | Requires large labeled datasets, may overfit | Classification, regression, fraud detection, NLP |
| **Unsupervised Learning** | Learns from unlabeled data to find patterns | - K-Means Clustering<br>- Hierarchical Clustering<br>- Principal Component Analysis (PCA)<br>- Autoencoders<br>- DBSCAN | Finds hidden structures, reduces dimensionality | No direct labels, harder to evaluate | Customer segmentation, anomaly detection |
| **Transfer Learning** | Uses a pre-trained model for a new task | - Fine-tuning Pretrained CNNs<br>- BERT (for NLP)<br>- GPT Models<br>- Domain Adaptation | Requires less data, faster training | May not work well for completely different tasks | Image recognition, NLP, medical diagnosis |
| **Reinforcement Learning** | Learns via trial & error using rewards | - Q-Learning<br>- Deep Q-Networks (DQN)<br>- Policy Gradient Methods<br>- Actor-Critic Methods | Works well in dynamic environments | Requires lots of training time, complex setup | Robotics, gaming AI, autonomous systems |
| **Self-Supervised Learning** | Uses unlabeled data to generate pseudo-labels | - Contrastive Learning<br>- SimCLR<br>- BYOL<br>- BERT (pretraining phase) | Learns from large datasets without labels | Computationally intensive | Large-scale AI models, feature learning |
| **Semi-Supervised Learning** | Uses a mix of labeled & unlabeled data | - Pseudo-labeling<br>- Self-training<br>- Graph-based Semi-Supervised Learning | Requires fewer labeled examples | Can propagate incorrect labels if mistakes occur | Medical imaging, speech recognition |

Would you like a deeper dive into a specific learning type or a practical example? 🚀

| **Method** | **Type** | **Strengths** | **Weaknesses** | **Best For** |
| --- | --- | --- | --- | --- |
| **Gaussian Discriminant Analysis (GDA)** | **Probabilistic, parametric** | **Captures feature correlations, provides probabilistic outputs** | **Assumes Gaussian distributions, can overfit with many features** | **Problems with normally distributed data, feature dependencies** |
| **Logistic Regression** | **Probabilistic, linear model** | **Simple, interpretable, efficient** | **Cannot handle complex non-linear boundaries** | **Binary classification, social sciences, medical studies** |
| **Linear Discriminant Analysis (LDA)** | **Probabilistic, parametric** | **Stable in high dimensions, interpretable** | **Assumes equal covariance across classes, less flexible than GDA** | **Linearly separable data, large feature spaces** |
| **Quadratic Discriminant Analysis (QDA)** | **Probabilistic, parametric** | **More flexible than LDA, allows distinct covariances per class** | **Requires more parameters, can overfit with limited data** | **When class distributions have different variances** |
| **Decision Trees** | **Non-parametric, rule-based** | **Easy to interpret, models non-linear boundaries** | **Prone to overfitting, sensitive to noise** | **When interpretability is key, handling mixed data types** |
| **Support Vector Machines (SVMs)** | **Non-parametric, geometric** | **Handles complex boundaries via kernels, good for high-dimensional data** | **Can be slow with large datasets, harder to interpret** | **High-dimensional feature spaces, complex decision boundaries** |
| **Neural Networks** | **Non-parametric, deep learning** | **Powerful for highly non-linear problems, learns intricate patterns** | **Requires large datasets, computationally intensive** | **Image recognition, speech processing, complex data** |

Imbalanced data

| Method | | Why It Works Well | | | Weaknesses | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Random Forest | | Uses multiple trees, reduces bias towards majority class | | | Can still struggle if class imbalance is extreme | | |
| Boosting (e.g., AdaBoost, XGBoost, LightGBM) | | Focuses on harder-to-classify examples, adjusts weights | | | Requires careful tuning to avoid overfitting | | |
| Support Vector Machines (SVMs) | | Finds a separation boundary with margin maximization | | | Can be sensitive to class imbalance without modifications | | |
| Logistic Regression with Class Weights | | Can assign higher weights to minority class to balance impact | | | Struggles with non-linear data unless engineered features are used | | |
| Synthetic Data Generation (SMOTE, ADASYN) | | Creates artificial minority samples to even out class distribution | | | Risk of overfitting if synthetic data is not generated properly | | |
| Anomaly Detection-Based Approaches | | Views minority class as an anomaly, detects deviations | | | Works best when the minority class is distinctly different from the majority class | | |
| **Method** | **Assumptions** | | **Decision Boundary** | **Strengths** | | **Weaknesses** | **Best Used For** |
| **Quadratic Discriminant Analysis (QDA)** | **Gaussian distributions per class, different variances** | | **Non-linear** | **Flexible, captures covariance** | | **Requires large data for accurate covariance estimates** | **Classification where classes have different spreads** |
| **Naïve Bayes** | **Assumes feature independence** | | **Linear (except in Gaussian NB)** | **Very fast, effective for text & simple problems** | | **Assumption of independence can be unrealistic** | **Text classification, spam filtering, quick predictions** |

**Key Strategies to Improve Classification on Imbalanced Data**

1. Resampling Methods – Oversampling the minority class (e.g., SMOTE) or undersampling the majority class.
2. Adjusting Class Weights – Some models (like Logistic Regression, SVMs, and Neural Networks) allow weight adjustments for class imbalance.
3. Using Ensemble Models – Methods like Boosting (XGBoost, LightGBM) help focus on difficult minority samples.

| **Metric** | **Type** | **Pros** | **Cons** | **Best Used For** |
| --- | --- | --- | --- | --- |
| **Accuracy** | **Classification** | **Simple, easy to interpret** | **Fails on imbalanced datasets** | **Balanced classification problems** |
| **Log Loss (Cross-Entropy Loss)** | **Classification** | **Evaluates probabilistic predictions** | **Over-penalizes wrong confident predictions** | **Probabilistic classification tasks** |
| **Precision** | **Classification** | **Minimizes false positives** | **Can reduce recall (miss true positives)** | **When false positives are costly (e.g., fraud detection)** |
| **Recall (Sensitivity)** | **Classification** | **Minimizes false negatives** | **Can reduce precision (more false alarms)** | **When false negatives are costly (e.g., medical diagnosis)** |
| **F1 Score** | **Classification** | **Balances precision & recall** | **Hard to interpret alone** | **Imbalanced datasets** |
| **ROC-AUC** | **Classification** | **Measures ranking ability** | **Can be misleading with highly skewed data** | **Binary classification, model comparison** |
| **Mean Squared Error (MSE)** | **Regression** | **Penalizes large errors** | **Sensitive to outliers** | **Predicting continuous values where large errors matter** |
| **Root Mean Squared Error (RMSE)** | **Regression** | **Same units as the target variable** | **Still penalizes large errors** | **Forecasting, financial predictions** |
| **Mean Absolute Error (MAE)** | **Regression** | **Treats all errors equally** | **Can be less sensitive to extreme errors** | **Predicting continuous values with equal error weighting** |
| **R² Score (Coefficient of Determination)** | **Regression** | **Explains variance in data** | **Hard to compare models with different datasets** | **Assessing model fit in regression problems** |
| **Mean Average Precision (MAP)** | **Ranking** | **Evaluates top results across queries** | **Less interpretable for non-ranking problems** | **Search engines, recommendation systems** |
| **Normalized Discounted Cumulative Gain (NDCG)** | **Ranking** | **Rewards ranking relevant results early** | **Needs relevance judgments for scoring** | **Ranking tasks like search engine results** |
| **Precision@K** | **Ranking** | **Evaluates top-K recommendations** | **Ignores lower-ranked results** | **Recommender systems** |
| **Silhouette Score** | **Clustering** | **Measures separation quality** | **Can be misleading with high-dimensional data** | **Evaluating cluster separation** |
| **Davies-Bouldin Index** | **Clustering** | **Evaluates cluster compactness** | **Lower values indicate better clustering** | **Comparing cluster quality** |
| **Adjusted Rand Index (ARI)** | **Clustering** | **Compares predicted clusters to ground truth** | **Only useful with known classifications** | **Validating clustering results** |
| **Dice Coefficient** | **Image Segmentation** | **Measures overlap between predicted and actual regions** | **Works best for binary segmentation tasks** | **Medical imaging, object detection** |
| **Mean Squared Logarithmic Error (MSLE)** | **Regression** | **Works well for exponential growth patterns** | **Hard to interpret for small-scale predictions** | **Financial forecasting, time series** |
| **OLS-F-statistic** | **Is the model useful** |  |  | **Whether at least one independent variable produces a significant model or not; larger the better** |
| **Log-likelhood** | **How well the model fits the data** |  |  | **Goodness of fit Higher the better** |
| **AIC** | **Helps to compare models** |  |  | **Lower =better balance between model fit and complexity** |
| **BIC-Bayesian information criterion** |  | **Penalizes complexity more than AIC** |  |  |
| **Precision** |  | **True positive/(True positive+ false Positives)** | **High chance of failure** |  |
| **Recall** | **True positive rate** | **True positive/(True positive+ false negatives)** | **Might miss some (base case – miss all)** |  |
| **F-1 statistic** | **Single metric for unbalanced data** |  |  |  |
| **Log loss** |  |  |  | **Lower log-loss indicates more accurate and confident predictions**  **(used when probability is important)** |
|  |  |  |  |  |

**..and use log-loss for tuning.**

**Choosing the Right Metric**

* If **false positives are costly** → Use **Precision**
* If **false negatives are costly** → Use **Recall**
* If **you need overall balance** → Use **F1 Score**
* If **you're predicting a continuous value** → Use **MSE, RMSE, or MAE**
* If **you're clustering data** → Use **Silhouette Score or ARI**
* If **you're working on search engines** → Use **NDCG or MAP**

**###Evaluate importance**

brown=default blue-not default

plot each 2x2plotvar 1 and var 2 using color by default or no default-and look for separation

**### Logistic regression**

Forces results between 0 and 1= p-hat(X)=e^linear reg/1+e^linear reg

log odds/logit -log (p(X))/1-p(X))=Beta0+beta1\*X

#4.2R-logit(.5) = Beta0+Beta1\*Balance;balance =(logit(.5)-beta0/beta1

(log(.5/(1-.5)) + 10.6513)/.0055 = 1936.6

-10.6\_=Coefficient for beta0 .00057=Coefficient for beta1

note: logit(.5)=(log(.5/(1-.5))

**#### R(glm);**

p-value chance that slope is zero (flat) i.e. between zero and one

estimate parameters - maximum Likelihood,ML(B0,B1)=joint probability of observing 0's,1's

**#R-code glm**

heartfit<-glm(chd~..,data=heart.family=binomial) . =all vars in dataframe; binomial -logitic regression model

**### Case control sampling-> need too correct the intercept**

Risk=0.05 % in age category=pie (**prior probability)**

sample -160 cases, 302 controls, risk =0.35=pie^

B0=B0+log(pie/1-pie)-log(pie^/1-pie^)

**### Unbalanced**

<0.005 or smaller

random sample

take a sample of controls

variance of parameter estimates vs case/control ratio -> diminishing returns 5:1 ratio

**### Discriminant**

Model the distribution of x in each class, then use bayes to flip things around

Use guassian distribution for each class

** Linear Discriminant Analysis (LDA) – Assumes that different groups have similar variance and finds a linear combination of features to best separate them.**

** Quadratic Discriminant Analysis (QDA) – Allows groups to have different variances, leading to more flexible boundaries between categories.**

**Bayes**

**PY=k/X=x) , joint distribution (multi-variable)**

**A math equation with red lines and black text

Description automatically generated**

**Prior probability = Pr (y=k0= piek**

**Marginal probability Pr(x=x)**

A math equations and formulas

Description automatically generated with medium confidence

**PDF for x in class k**

**Marginal p/x = summing over all the classes**

**Plug in Gaussian density**

**A diagram of a function

Description automatically generated**

**Different priors in second, move boundary to the left**

**If classes are well separate , parameters for logistic regression are unstable**

**If sample size is small, and distribution is normal, discriminant model is more stable.**

If ad clicks are low (qualitative variable – doesn’t have a density), the ad clicks have a low probability.

Prior probability- likelihood of an outcome based on prior knowledge or general reasoning rather than specific observations or data

**### Three variable classification/ multi-class log or**

**#More than two classes**

Glmnet package

A screenshot of a computer screen

Description automatically generated

Each class gets a linear model and weight them against each other with **Softmax function**

**Gaussian Discriminant Analysis - One Variable**

Guassian density function for class k for a single xA math equation with a red line

Description automatically generated

**Simplication - see which is the largest probability to classify**

A math equations on a white paper

Description automatically generated with medium confidence

X-single variable

Mean

Varian

Prior

Single coofifient for x

A math equations and formulas

Description automatically generated with medium confidenceGet a function for each class

If K=2 classes, **decision boundary is x mu1+mu2/ 2**

**Estimate means and common standard deviations**

**A graph and diagram of a function

Description automatically generated**

**A screenshot of a math equation

Description automatically generated**

**Pi,k=priors**

**TO determine the decision boundary**

**Gaussian Discriminant Analysis - Many Variables**

Gaussian Discriminant Analysis (GDA) is a probabilistic classification method based on Bayes’ theorem, assuming that the data for each class is drawn from a Gaussian distribution. When dealing **with many variables, GDA works in high-dimensional feature spaces**, making it an effective tool for separating complex datasets.

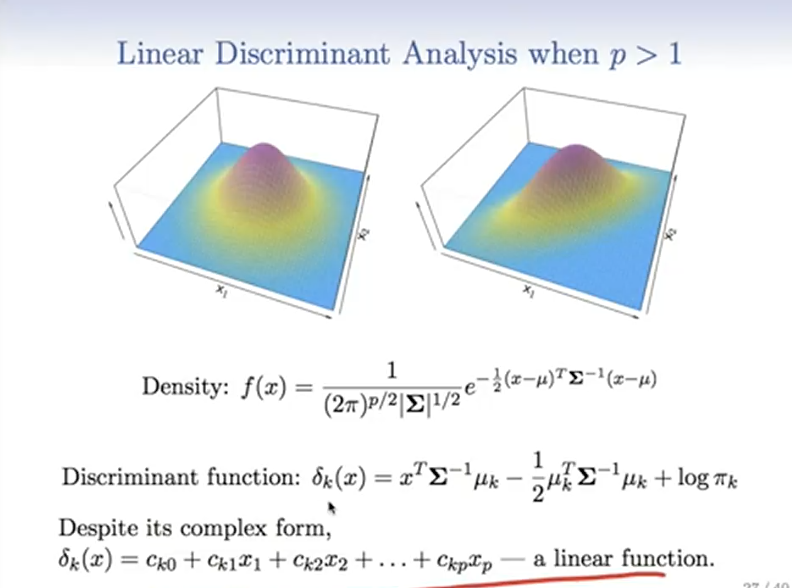
Here's how it works:

1. Assumption of Gaussian Distribution – Each class in the dataset is modeled as a multivariate normal (Gaussian) distribution. The mean vector and covariance matrix for each class are estimated from the training data.
2. Bayes' Theorem for Classification – A new observation is classified based on its likelihood under each class’s estimated Gaussian distribution.
3. Linear or Quadratic Boundaries – If the covariance matrices for all classes are assumed to be the same, GDA results in a linear decision boundary (this is equivalent to Linear Discriminant Analysis). **If each class has its own covariance matrix, the boundary is quadratic, making it** more flexible but requiring more **parameters.**

Why It Works Well for Many Variables:

* **Handles Correlations Between Features** – The covariance matrix encodes relationships between multiple variables, **allowing GDA to capture complex dependencies.**
* Bayesian Nature – It provides a probabilistic interpretation, **which can be useful when working with uncertain or noisy data**.
* Dimensionality Considerations – While GDA scales to multiple variables, estimating covariance matrices can become challenging in very high-dimensional spaces, where regularization techniques may be needed to avoid overfitting.

**Gaussian density with 2 variables**

****

**Linear in x, all others are constant**

**Left – no correlation between x1, x2**

**Right – skewed i.e. correlation between x1 and x2**

**Two variables,3 classes**

**A screenshot of a graph

Description automatically generated**

**A data analysis chart with different colors

Description automatically generated with medium confidence**

**4 variables**

**Linear combinations of variable-Guassian LDA (which centroid is closest?) A diagram of a number of colored dots

Description automatically generated with medium confidence**

**4,000 (covariance matrix-4kx4k)**

**DA with probabilities**

**k-2 >50% =class 2, <50% class 1**

**A screenshot of a math book

Description automatically generated**

**Small dataset-separate test setA screenshot of a credit data

Description automatically generated**

**Classified to the prior /null rate=3.33%**

True no=

True yes=

False positive rate=2%

False negative=75.7

Change threshold i.e. make smallerA graph of negative rate

Description automatically generated

Capture changing threshold with ROC curve for all possible thresholds

False positive to be low=0, True positive to be high=1

A screen shot of a graph

Description automatically generated

AUC=area under the curve, higer is good

Machine learning/learning rate

==

Binary-logit

Linear-quantitative response

Non-negative

Skewed distributions

A screenshot of a computer screen

Description automatically generated

A graph of a number of numbers and a line

Description automatically generated with medium confidence

P1- Variance increasing with the mean

Assume variance is constant with linear regressions

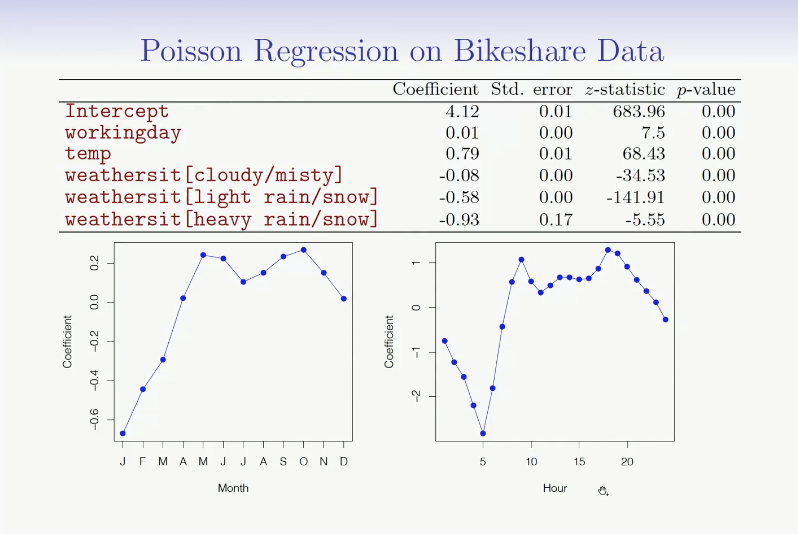
Fit model,10% are negative

P2-can take logs of zero, don’t need logs on independent

Poisson Regression good for modeling counts-mean higher, variance is higher

Binomial mean=p, variance is P times 1 minus P

Poisson regression on log scale



p-values areas are misleading small, account for by accommodating the overdispersion

GLM – link function

Transformation of the mean that is a linear model

A math equations on a white background

Description automatically generated

Linear-identity link/quantitiave (Gaussian-symmetric)

Logistic – logit link

Poisson – log link (counts)

Binomial – binary (Binary)

(nature of response)

Gamma – positive observations, long-tails to the write

Negative-binomial/ fixing overdispersion (counts)

Inverse gaussian

4.8.R1

Which of the following statements is true for linear logistic regression?

The probability of a binary outcome is represented as a linear function of the the input features.

The model specifies a form for the mean as well as the variance of the response.

The model forces the modeled mean to lie between zero and one.

Explanation

The logit of the mean is linear in the features.

If the mean is  then the variance is given by .A black line and a black line

Description automatically generated

The inverse logit transform is is   which lies between 0 and 1 by construction.

The model is fit bt maximum-likelihood, which is often implemented as "iteravtively-reweighted least squares".

**Quadratic Discriminant Analysis and Naive Bayes**

Quadratic Discriminant Analysis (QDA) is a classification algorithm based on Bayes’ theorem with the assumption that each class follows a Gaussian (normal) distribution with its own covariance matrix

when classes have **distinct variances**, but can overfit when sample sizes are small.

Best Used For:

* Cases where different classes have **varying spreads** in feature space.
* When a **non-linear decision boundary** is needed but data still follows a Gaussian distribution.

General discriminant analysis

**A math equation with black text

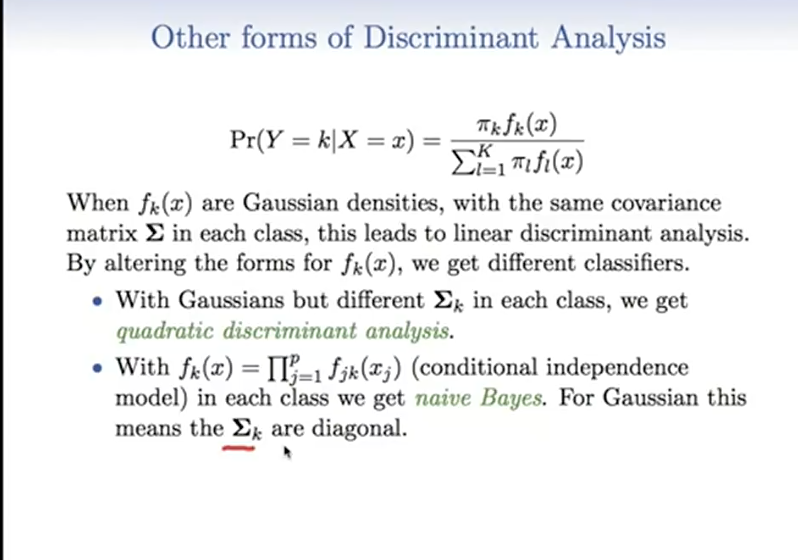
Description automatically generated with medium confidence**

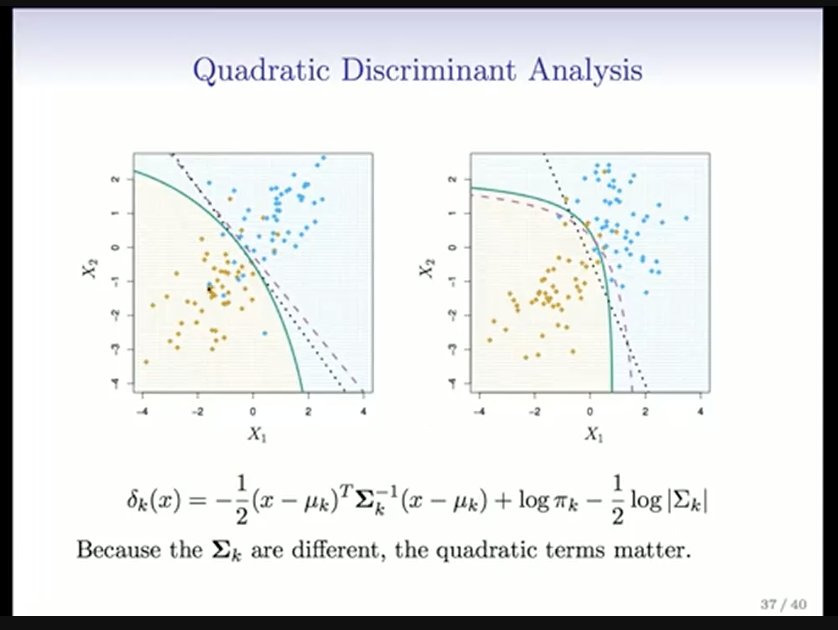
Gaussian densities

Use other density estimates to get classification rules

**If variances are different, the quadratic terms don’t cancel-quadratic descriminant analysis**

**Large number of features** (don’t want to estimate large If assume density factors into a product of densities; i.e. the vars are conditionally depended =**Naïve Bayes classifier.**



Different

Distance term=sigma sub kth class + Prior prob – log piek. +. Determinant term from covariance matrix

Dotted- linear discriminatory analysis (large # of vars can break down)

Line-dotted- Quad discriminant analyses (# of variables is small)

Solid- Bayes

Different covariance matrix for each class.

**Naive Bais – lots on variables, Can use mixed features (qualitative-histograms PMFs and quantitative-gaussian)**

| **Feature** | **Discriminant Analysis** | **Logistic Regression** |
| --- | --- | --- |
| **Underlying Model** | Assumes data follows a Gaussian distribution within each class | Models probability directly using a logistic (sigmoid) function |
| **Assumption of Data** | Requires normality and equal covariances (LDA); relaxed in QDA | Does not assume normality or equal covariances |
| **Decision Boundary** | Can be **linear (LDA)** or **quadratic (QDA)** | Always **linear** unless non-linear terms are added |
| **Estimation Approach** | Uses **Bayes' theorem** to compute probabilities | Uses **maximum likelihood estimation (MLE)** |
| **Robustness** | More efficient with **small sample sizes** when assumptions hold | More flexible, works well even with non-Gaussian distributions |
| **Interpretability** | Provides class probabilities based on distribution | Directly interprets impact of each feature on probability |
| **Best Used For** | When data distribution follows Gaussian assumptions | When relationships between features and classes are **not** normally distributed |
|  | Same form-linear logistic models |  |
|  | Estimating parameters using full likehood-full distribution of x and y, **generative learning** | Conditional likelihood based no P(Y| X), **discriminative learning**  Used full distribution of y |
|  | Quadratic boundaries | Quadratic boundaries, include quadratic terms in the model |

Which of the following statements best explains the relationship between Quadratic Discriminant Analysis and naive Bayes with Gaussian distributions in each class?

Quadratic Discriminant Analysis is a more flexible class of models than naive Bayes.

==