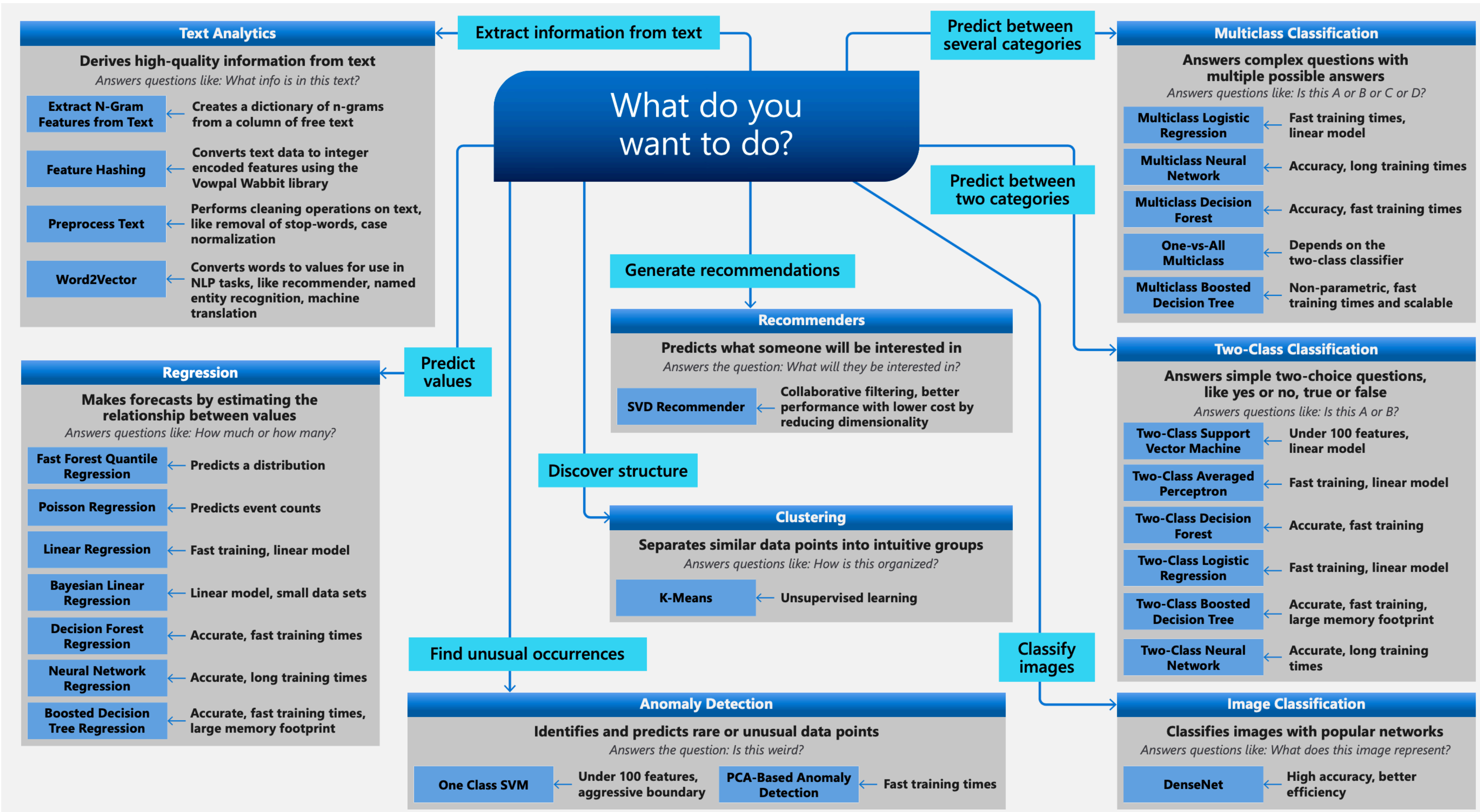
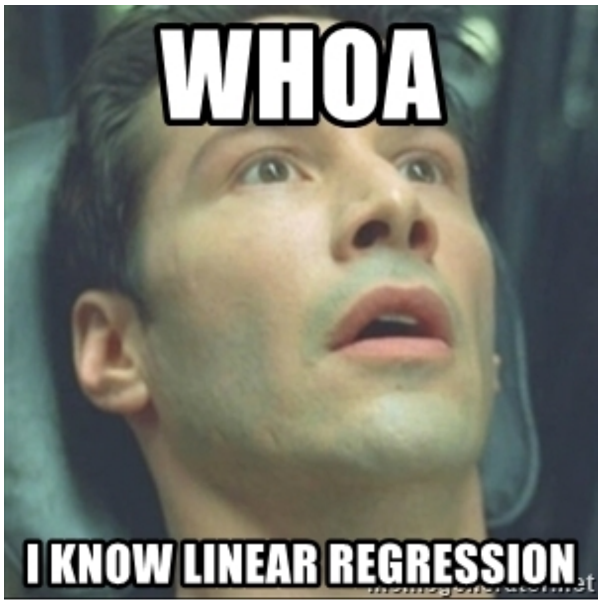
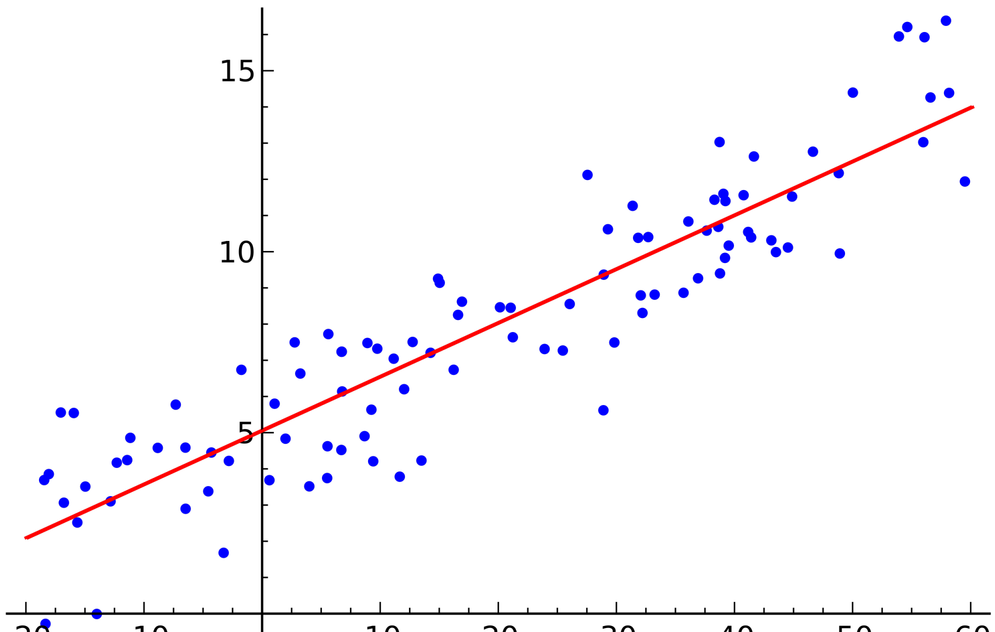

What do you want to do?
Extract information from text
Text Analytics
Derives high-quality information from text
Answers questions like: What info is in this text?
Extract N-Gram Features from Text
Creates a dictionary of n-grams from a column of free text
Feature Hashing
Converts text data to integer encoded features using the Vowpal Wabbit library
Preprocess Text
Performs cleaning operations on text, like removal of stop-words, case normalization
Word2Vector
Converts words to values for use in NLP tasks, like recommender, named entity recognition, machine translation
Predict values
Regression
Makes forecasts by estimating the relationship between values
Answers questions like: How much or how many?
Fast Forest Quantile Regression
Predicts a distribution
Poisson Regression
Predicts event counts
Linear Regression
Fast training, linear model
Bayesian Linear Regression
Linear model, small data sets
Decision Forest Regression
Accurate, fast training times
Neural Network Regression
Accurate, long training times
Boosted Decision Tree Regression
Accurate, fast training times, large memory footprint
Generate recommendations
Recommenders
Predicts what someone will be interested in
Answers the question: What will they be interested in?
SVD Recommender
Collaborative filtering, better performance with lower cost by reducing dimensionality
Discover structure
Clustering
Separates similar data points into intuitive groups
Answers questions like: How is this organized?
K-Means
Unsupervised learning
Find unusual occurrences
Anomaly Detection
Identifies and predicts rare or unusual data points
Answers the question: Is this weird?
One Class SVM
Under 100 features, aggressive boundary
PCA-Based Anomaly Detection
Fast training times
Predict between several categories
Multiclass Classification
Answers complex questions with multiple possible answers
Answers questions like: Is this A or B or C or D?
Multiclass Logistic Regression
Fast training times, linear model
Multiclass Neural Network
Accuracy, long training times
Multiclass Decision Forest
Accuracy, fast training times
One-vs-All Multiclass
Depends on the two-class classifier
Multiclass Boosted Decision Tree
Non-parametric, fast training times and scalable
Predict between two categories
Two-Class Classification
Answers simple two-choice questions, like yes or no, true or false
Answers questions like: Is this A or B?
Two-Class Support Vector Machine
Under 100 features, linear model
Two-Class Averaged Perceptron
Fast training, linear model
Two-Class Decision Forest
Accurate, fast training
Two-Class Logistic Regression
Fast training, linear model
Two-Class Boosted Decision Tree
Accurate, fast training, large memory footprint
Two-Class Neural Network
Accurate, long training times
Classify images
Image Classification
Classifies images with popular networks
Answers questions like: What does this image represent?
DenseNet
High accuracy, better efficiency



Linear Regression

Predicting Values

Simple* Linear Regression



* Like, mathematicians refer to linear regression with one independent variable as “simple”, not a comment on linear regression actually *being* simple

Chalkboard Example

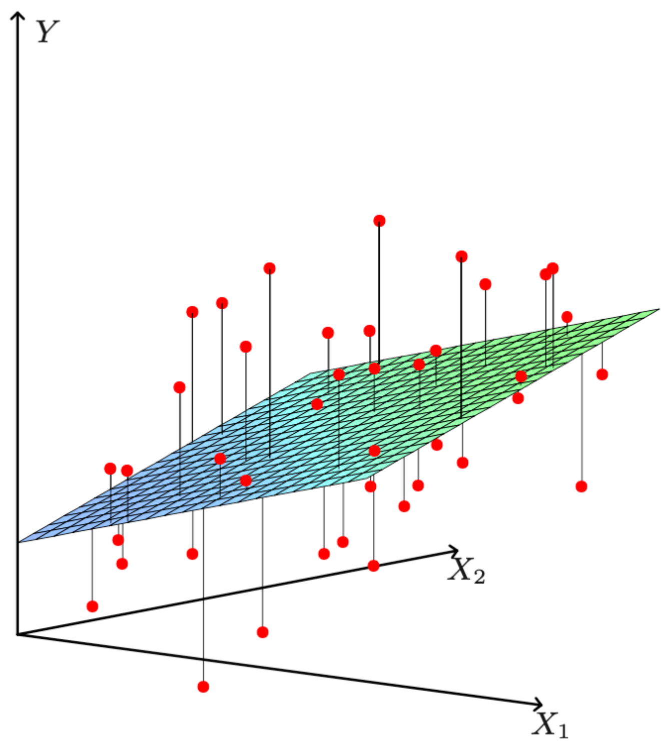


FIGURE 3.1. *Linear least squares fitting with* $X \in \mathbb{R}^2$. *We seek the linear function of* X *that minimizes the sum of squared residuals from* Y.

General Linear Regression Model

We have a vector of p attributes, $X = (X_1, X_2, \dots, X_p)$

We want to use these p attributes to predict a value, $f(X)$

We find $p + 1$ values for beta $(\beta_0, \beta_1, \beta_2, \dots \beta_p)$ based on our observed data.

$$f(X) = \beta_0 + \sum_{j=1}^{p} X_j \beta_j$$

Finding Beta Hat

- Step 1 : Build a matrix $\boldsymbol{X}$ which will have dimensions m x (p + 1), where m is the number of observations in the dataset and p is the number of <u>attributes</u>.

$$\boldsymbol{X} = \begin{bmatrix} 1 & x_{1,1} & x_{1,2} & \dots & x_{1,p} \\ 1 & x_{2,1} & x_{2,2} & \dots & x_{2,p} \\ 1 & x_{2,1} & x_{3,2} & \dots & x_{3,p} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{m,1} & x_{m,2} & \dots & x_{m,p} \end{bmatrix}$$

Finding Beta Hat

- Step 2 : Build the vector $\boldsymbol{y}$ which will have dimensions m x 1 that represents past examples of the value you want to predict based on the observations in the $\boldsymbol{X}$ matrix

$$\boldsymbol{y} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_m \end{bmatrix}$$

Finding Beta Hat

- Step 3 : Find the pseudo inverse of the $\boldsymbol{X}$ matrix and then multiply by the vector $\boldsymbol{y}$. It will have dimensions (p + 1) x 1,

$$\hat{\beta} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{y}$$

What *is* Beta Hat?

- Beta hat is a vector containing the least squares solution to the general linear regression model.

- This vector can now be used to predict any value given a set of p attributes.

- This is “The Model”

Find a Prediction, y hat

- Assume we have a new observation $\dot{x} = (\dot{x}_1, \dot{x}_2, \dots, \dot{x}_p)$, we create the row vector of size 1 x (p + 1)

$$\dot{\boldsymbol{x}} = \begin{bmatrix} 1 & \dot{x}_1 & \dot{x}_2 & \dots & \dot{x}_p \end{bmatrix}$$

- Find a predicted value. The result, y hat, is a single value.

$$\hat{y} = \dot{\boldsymbol{x}}\hat{\beta}$$