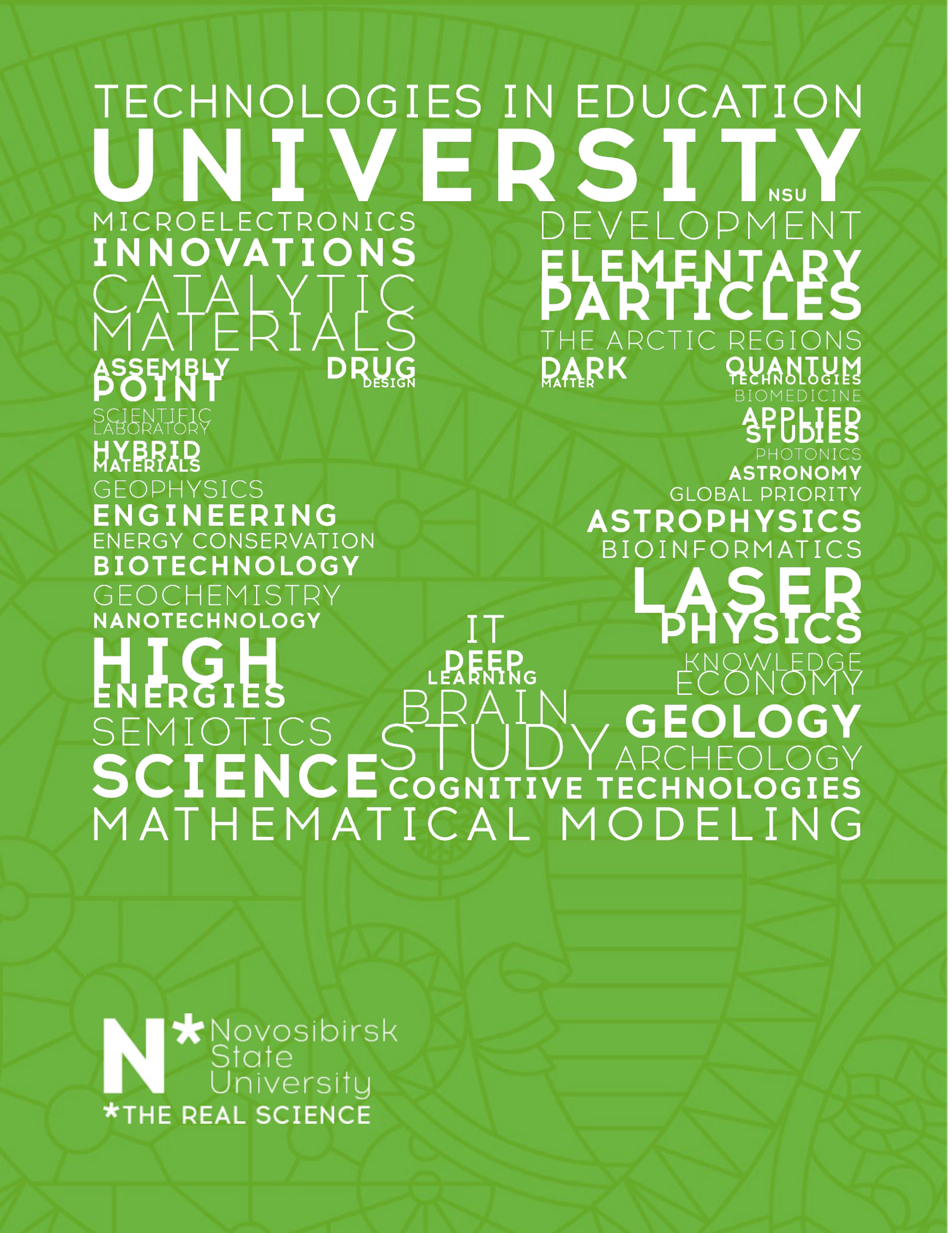


Logistic Regression

Luu Minh Sao Khue
Department of Mathematics and Mechanics

Overview

- Classification in Machine Learning
- Logistic Regression
- Evaluation metrics
- K-nearest Neighbor

Classification in Machine Learning

Regression vs Classification

Regression	Classification
outcome is continuous (numerical)	outcome is a category

Prediction examples:
- House prices
- Box office revenues
- Event attendance
- Network load
- Portfolio losses

Prediction examples:
- Detecting fraudulent transactions
- Customer churn
- Event attendance
- Network load
- Loan default

Classification

Question	Answer "y"	
Is this email spam?	no	yes
Is the transaction fraudulent?	no	yes
Is the tumor malignant?	no	yes

y can only be one of two values

"binary classification"

Classification

Question	Answer "y"	
Is this email spam?	no	yes
Is the transaction fraudulent?	no	yes
Is the tumor malignant?	no	yes

y can only be one of two values

"binary classification"

class = category

false	true
0	1

useful for classification

"negative class" ≠ "bad" absence

"positive class" ≠ "good" presence

Classification

Question	Answer "y"	
Is this email spam?	no	yes
Is the transaction fraudulent?	no	yes
Is the tumor malignant?	no	yes
	false	true
	0	1

useful for classification

y can only be one of two values

"binary classification"

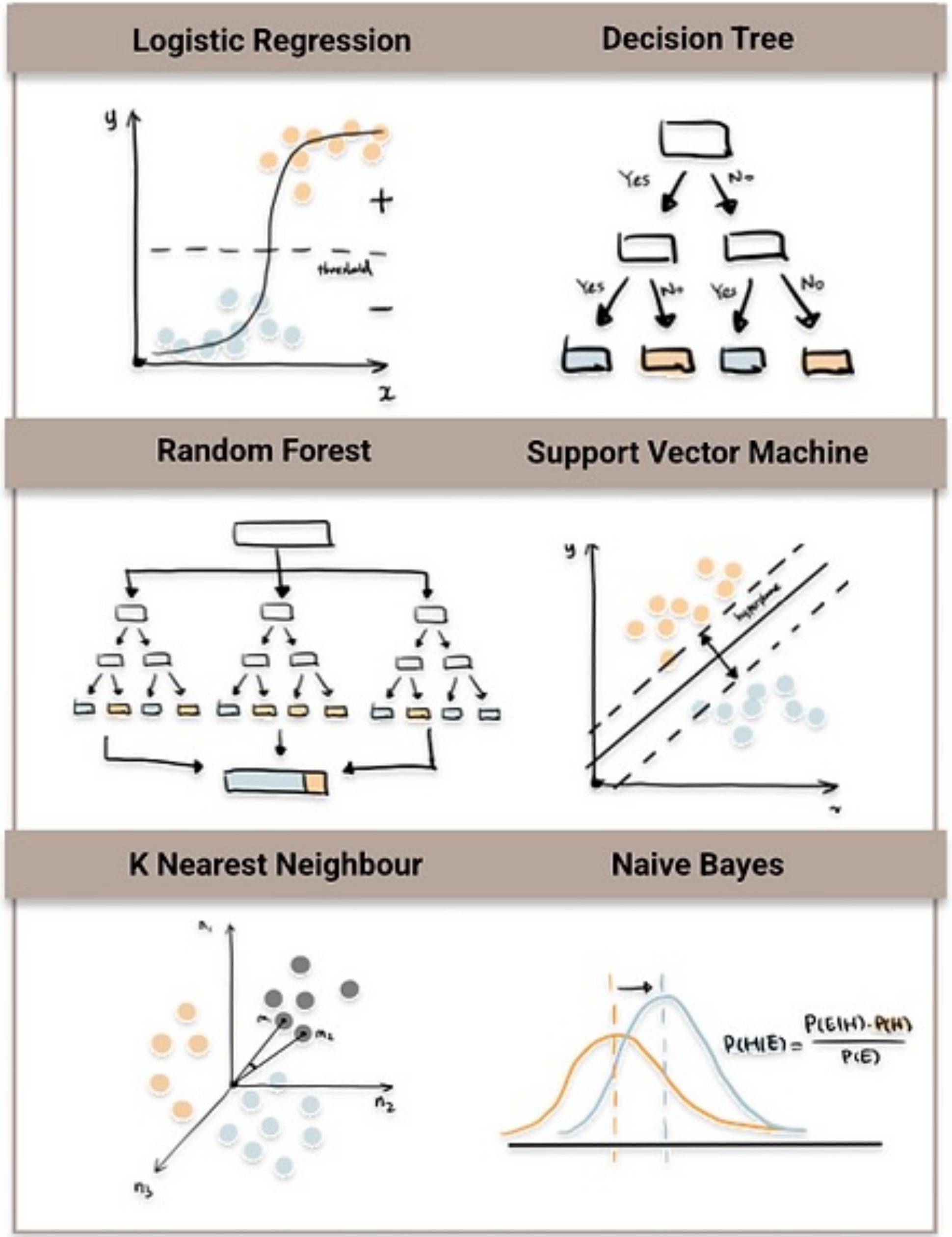
class = category

"negative class" ≠ "bad" absence

"positive class" ≠ "good" presence

Classification Algorithms

- **Linear Models**:
  - Logistic Regression
  - Support Vector Machines (SVM)
- **Instance-Based Models**
  - k-Nearest Neighbors (kNN)
- **Tree-Based Models**
  - Decision Trees
  - Random Forest
  - Gradient Boosting (XGBoost, LightGBM, CatBoost)
- **Probabilistic Models**
  - Naive Bayes
- **Neural Networks**
  - Deep Learning for complex, high-dimensional data.



Source: https://www.visual-design.net/post/top-machine-learning-algorithms-classification

Eager Learning vs Lazy Learning

- An **eager learner** is a model that generalizes the data into a fixed function or model during the training phase. It builds a comprehensive model based on the entire training dataset before making predictions.
- Characteristics:
  - The model does all the work during training.
  - Once trained, predictions are fast because the model doesn't need the original data.
  - Generalizes patterns in the data into a global model.
  - Requires retraining when new data is added.
- Examples: Logistic Regression, Decision Tree, Neural Network, etc.

Eager Learning vs Lazy Learning

- Advantages:
  - Faster prediction as the model is ready after training.
  - Better for applications with strict latency requirements (e.g., real-time systems).
  - Works well with large datasets during training.
- Disadvantages:
  - Time-consuming training phase, especially with complex models.
  - Not flexible for incremental learning (new data requires retraining).

Eager Learning vs Lazy Learning

- A **lazy learner** does not build a comprehensive model during training. Instead, it stores the training data and processes it only when making predictions. Lazy learners memorize the data and defer the generalization until a query (prediction) is made.
- Characteristics:
  - Minimal or no work during training.
  - Predictions are computationally intensive because the model must access and process training data.
  - Local models are created dynamically for each prediction.
- Example: k-nearest neighbor

Eager Learning vs Lazy Learning

- Advantages:
  - Simple and easy to implement.
  - No retraining is required when new data is added (incremental learning).
  - Works well for datasets with irregular or complex patterns.
- Disadvantages:
  - Slow predictions, especially with large datasets, because it involves searching through the training data.
  - Requires storing the entire dataset, leading to high memory usage.
  - Sensitive to irrelevant or noisy features.

Eager Learning vs Lazy Learning

Aspect	Eager Learning	Lazy Learning
Training Phase	Builds a model during training	Stores data, no model built upfront
Prediction Phase	Fast (uses the pre-built model)	Slow (accesses and processes data)
Generalization	Global (learns from the entire dataset)	Local (focuses on the query instance)
Flexibility	Requires retraining for new data	Can incorporate new data directly

Logistic Regression

Logistic Regression

Independent variables | Dependent variable

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	Yes
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	Yes
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	No
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	No
4	7.0	35.0	14.0	80.0	2.0	15.0	0.0	1.0	0.0	?

Categorical Variable

Continuous/Categorical variables

Logistic Regression

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	1.0
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	1.0
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	0.0
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	0.0

$X \in \mathbb{R}^{m \times n}$

$y \in \{0,1\}$

$$\boxed{\hat{y} = \mathrm{P}(y=1|x)}$$

$\mathrm{P}(y=0|x) = 1 - \mathrm{P}(y=1|x)$

Logistic Regression

$$\overbrace{\text{tenure} \quad \text{age} \quad \text{address} \quad \text{income} \quad \text{ed} \quad \text{employ} \quad \text{equip} \quad \text{callcard} \quad \text{wireless}}^{\mathbf{X}} \quad \overbrace{\text{churn}}^{y}$$

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	1.0
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	1.0
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	0.0
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	0.0

$X \in \mathbb{R}^{m \times n}$

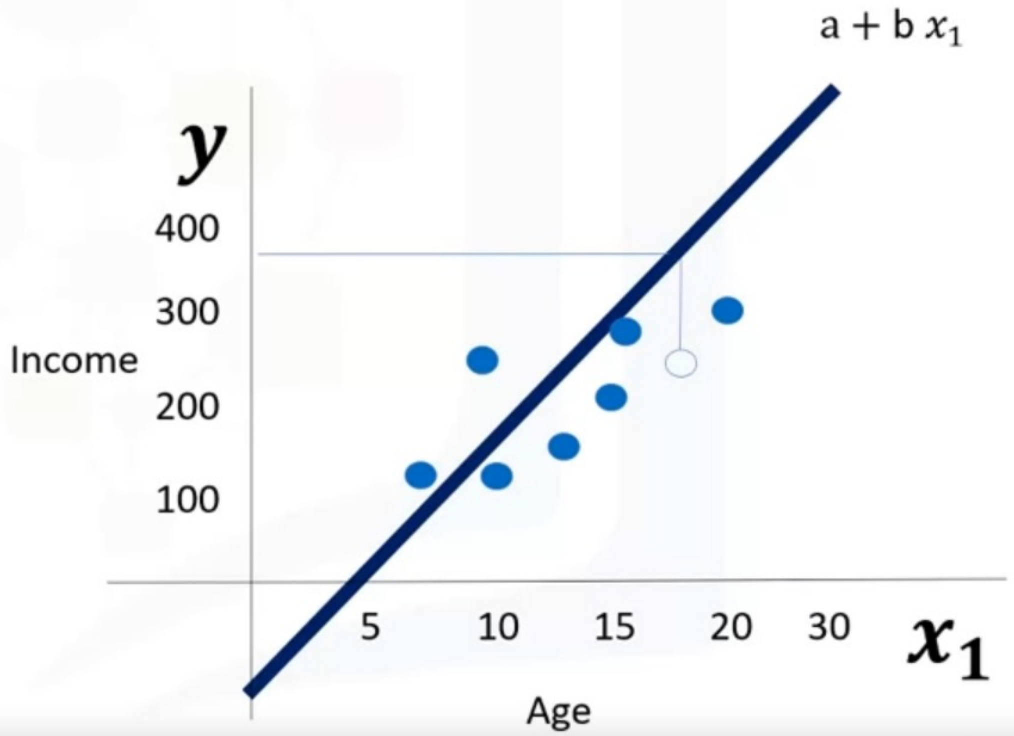
$y \in \{0,1\}$

$$\boxed{\hat{y} = P(y=1|x)}$$

$P(y=0|x) = 1- P(y=1|x)$

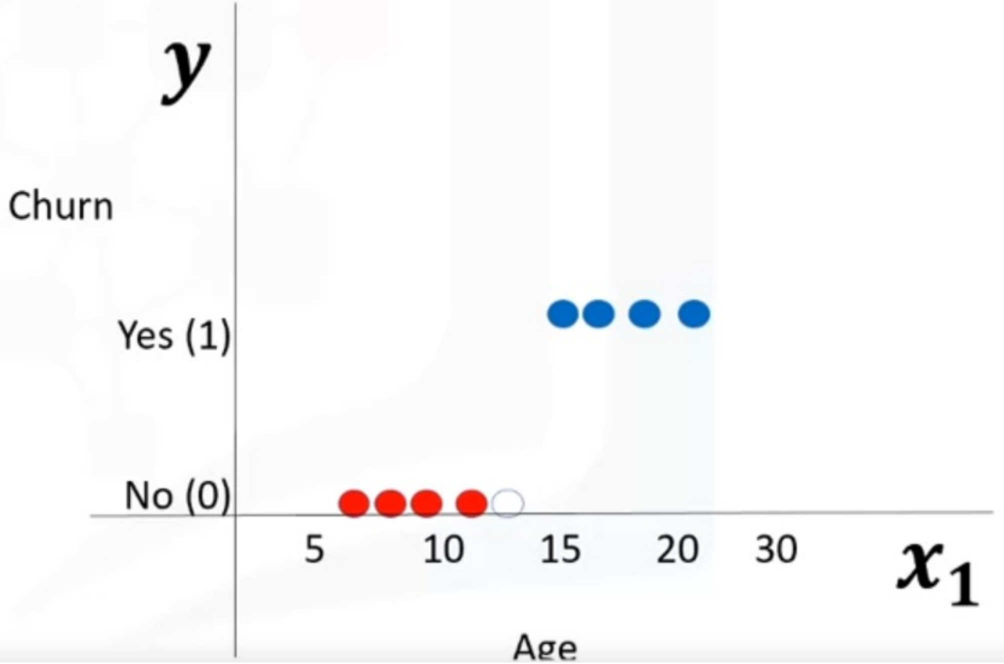
Linear Regression vs Logistic Regression

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	1
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	1
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	0
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	0
4	7.0	35.0	14.0	80.0	2.0	15.0	0.0	1.0	0.0	0



Linear Regression vs Logistic Regression

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	1
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	1
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	0
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	0
4	7.0	35.0	14.0	80.0	2.0	15.0	0.0	1.0	0.0	0



Linear Regression vs Logistic Regression

$\theta^T X = \theta_0 + \theta_1 x_1$

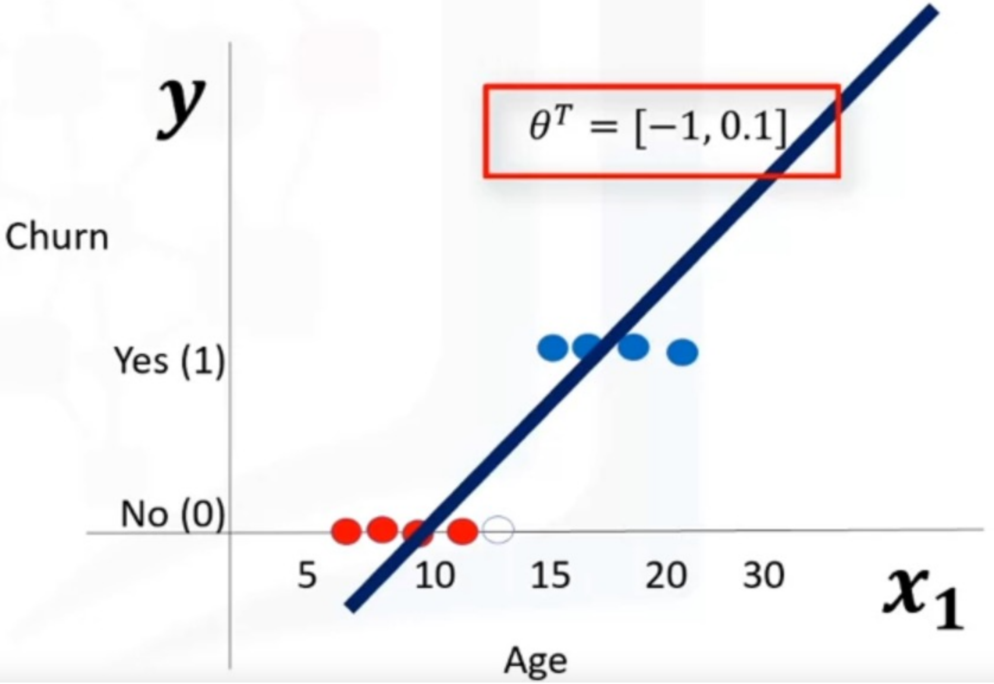
$\theta^T X = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \ \ldots$

$\theta^T = [\theta_0, \theta_1, \theta_2, \ldots]$ $\quad X = \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ \ldots \end{bmatrix}$

$\theta^T = [\theta_0, \theta_1]$

$\theta_0 + \theta_1 x_1$

$\mathrm{a} + \mathrm{b}\, x_1$



Linear Regression vs Logistic Regression

$\theta^T X = \theta_0 + \theta_1 x_1$

$\boldsymbol{p}_1 = [13] \rightarrow \theta^T X = -1 + 0.1 \,.\, x_1$

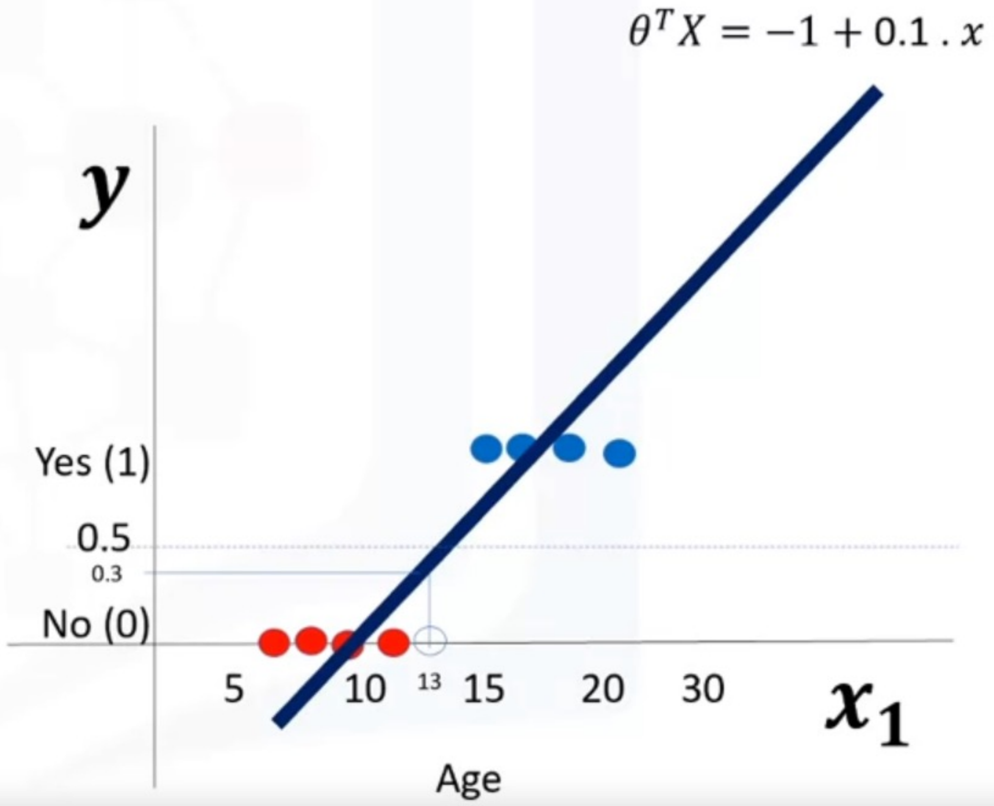
$= -1 + 0.1 \times 13$

$= 0.3$

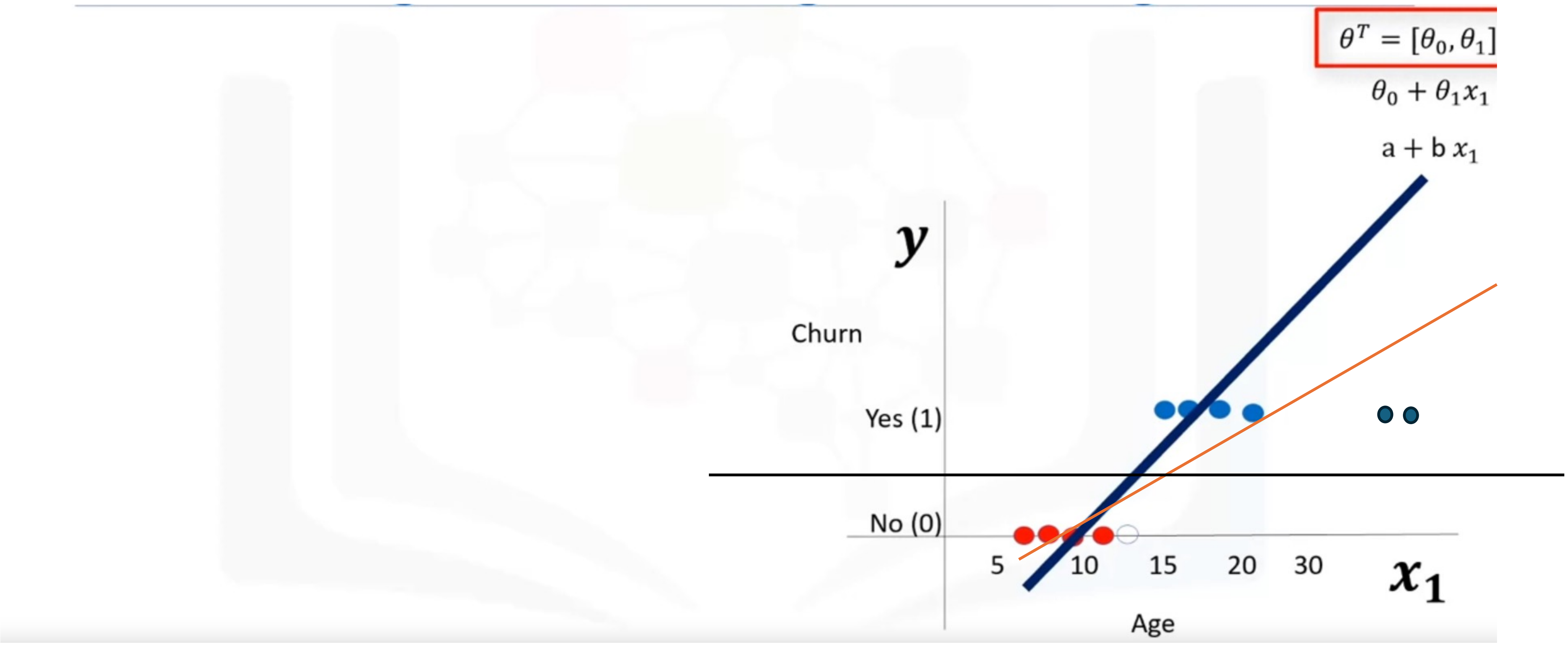
$$\hat{y} = \begin{cases} 0 & \text{if } \theta^T X < 0.5 \\ 1 & \text{if } \theta^T X \geq 0.5 \end{cases}$$

$\theta^T X = 0.3$

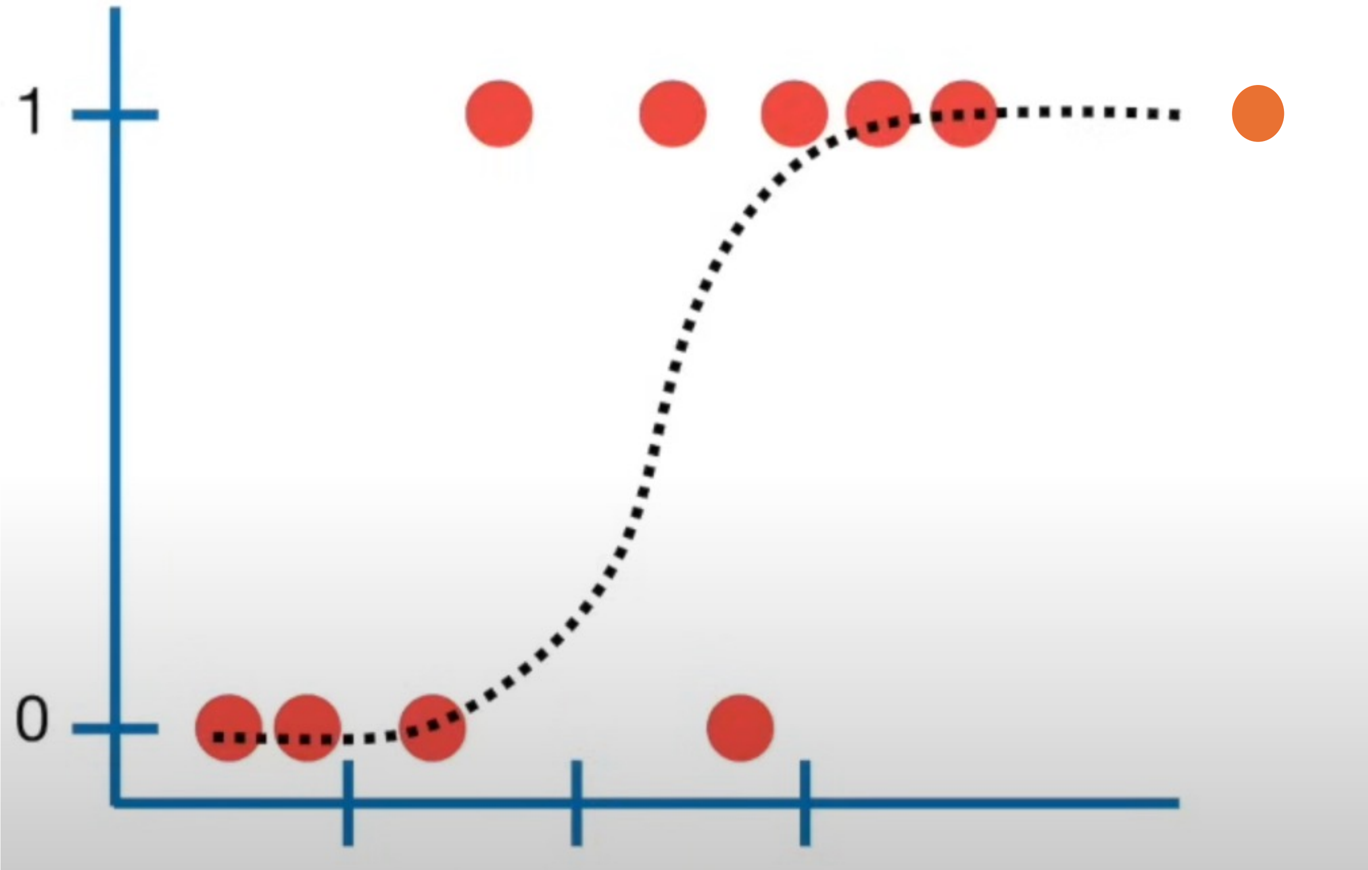
$\theta^T X < 0.5 \rightarrow$ Class 0



Linear Regression vs Logistic Regression

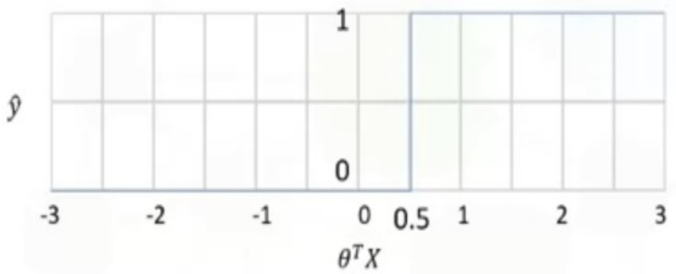


Linear Regression vs Logistic Regression



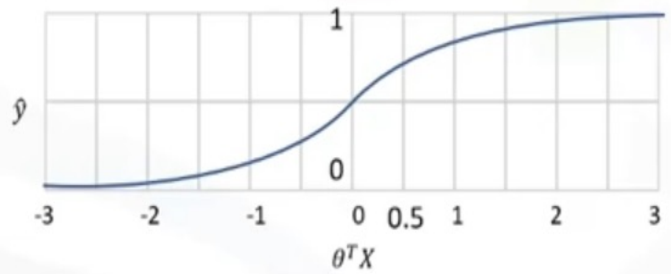
Linear Regression vs Logistic Regression

$$\theta^T X = \theta_0 + \theta_1 x_1 + \cdots$$



$$\hat{y} = \begin{cases} 0 & \text{if } \theta^T X < 0.5 \\ 1 & \text{if } \theta^T X \geq 0.5 \end{cases}$$

$$\sigma(\theta^T X) = \sigma(\theta_0 + \theta_1 x_1 + \cdots)$$



$$\boxed{\hat{y} = \sigma(\theta^T X)}$$

$\boxed{\text{P(y=1|x)}}$

Training steps

$$\sigma(\theta^T X) \longrightarrow P(y=1|x)$$

1. Initialize θ.
2. Calculate $\hat{y} = \sigma(\theta^T X)$ for a customer.
3. Compare the output of $\hat{y}$ with actual output of customer, y, and record it as error.
4. Calculate the error for all customers.
5. Change the θ to reduce the cost.
6. Go back to step 2.

$\theta = [-1,2]$

$\hat{y} = \sigma([-1,2] \times [2,5]) = 0.7$

Error = 1-0.7 = 0.3

$Cost = J(\theta)$

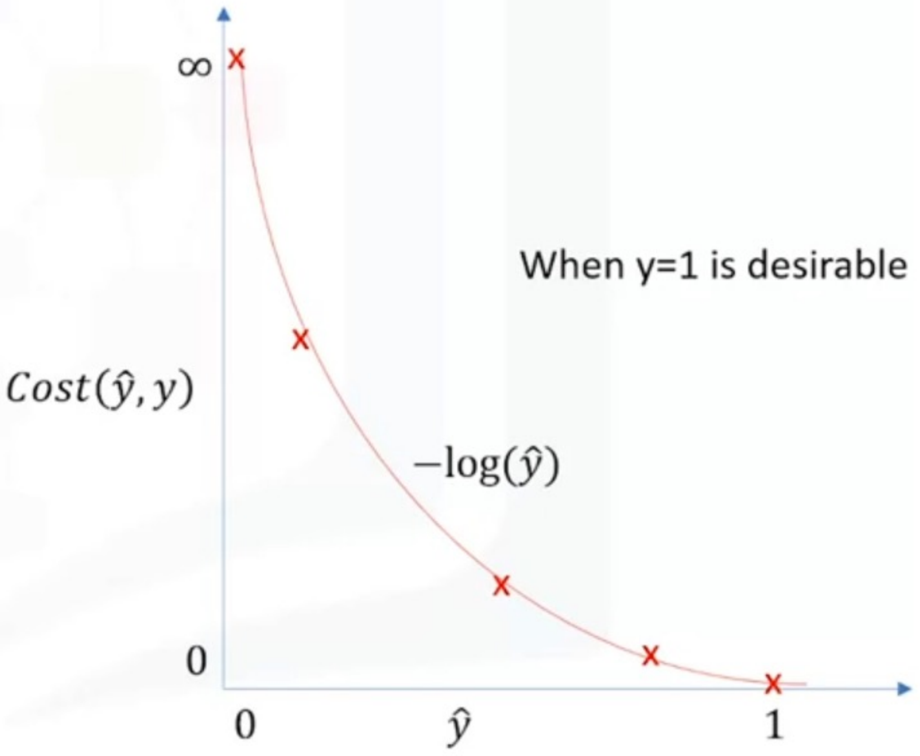
θ_{new}

Loss function

- Model $\hat{y}$
- Actual Value y=1 or 0

- If Y=1, and $\hat{y}$=1 → cost = 0

- If Y=1, and $\hat{y}$=0 → cost = large



Loss function

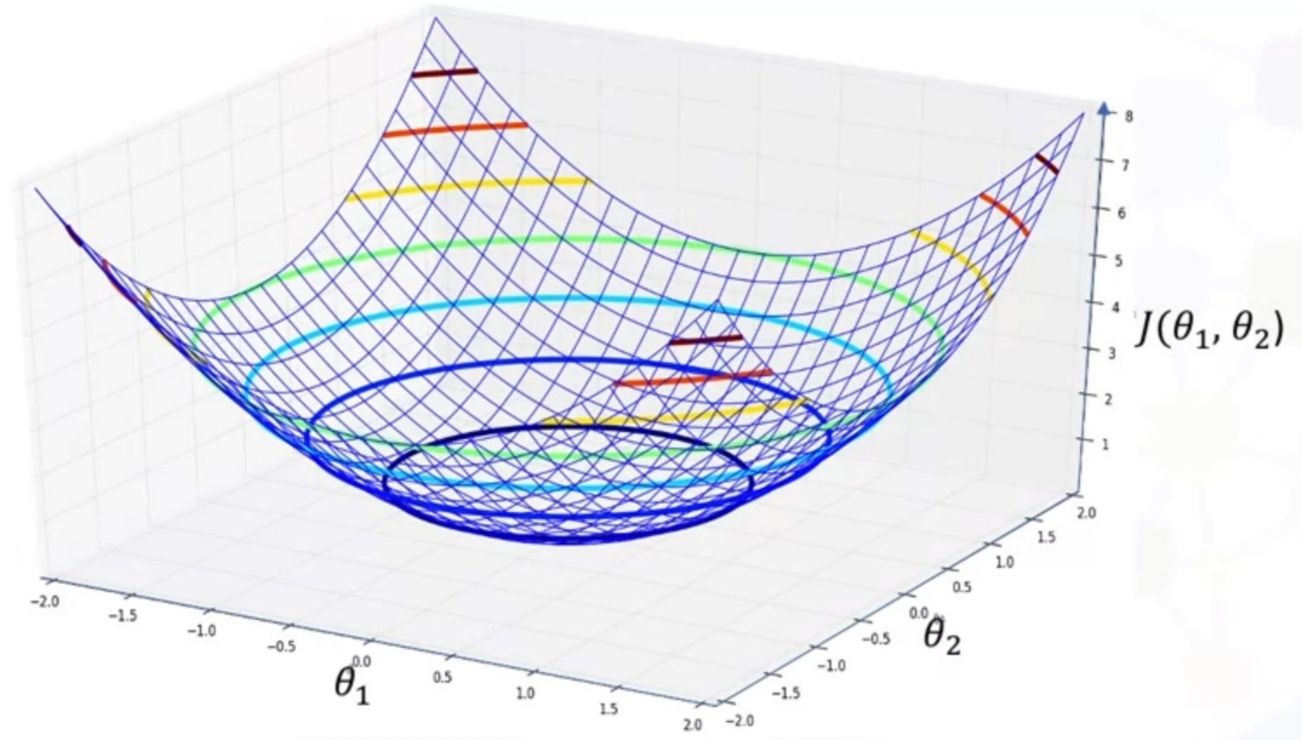
$$\cancel{Cost(\hat{y}, y) = \frac{1}{2}\left(\sigma\left(\theta^T X\right) - y\right)^2}$$

$$Cost(\hat{y}, y) = \begin{cases} -\log(\hat{y}) & if\ y = 1 \\ -\log(1 - \hat{y}) & if\ y = 0 \end{cases}$$

$$\cancel{J(\theta) = \frac{1}{m}\sum_{i-1}^{m} Cost(\hat{y}, y)}$$

$$\boxed{J(\theta) = -\frac{1}{m}\sum_{i-1}^{m} y^i \log\left(\hat{y}^i\right) + (1 - y^i)\log\left(1 - \hat{y}^i\right)}$$

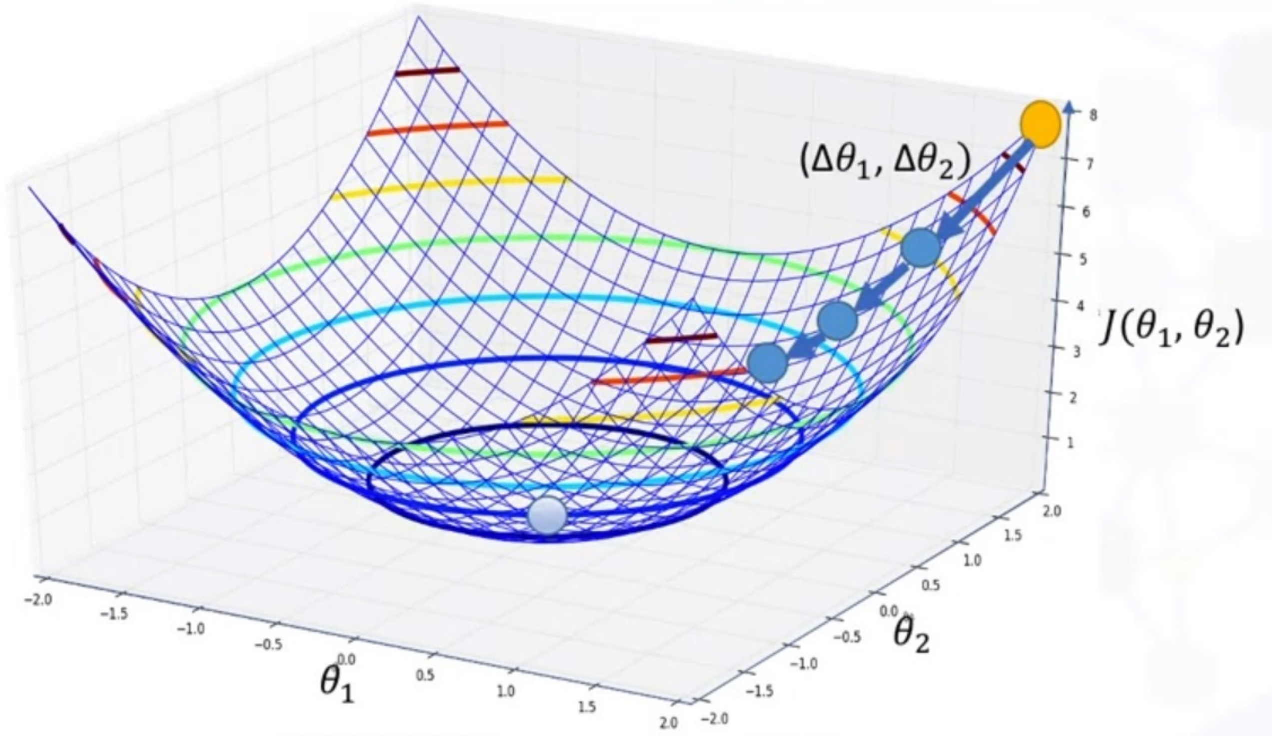
Minimizing loss with gradient descent



$\hat{y} = \sigma(\theta_1 x_1 + \theta_2 x_2)$

$$J(\theta) = -\frac{1}{m}\sum_{i-1}^{m} y^i \log(\hat{y}^i) + (1 - y^i)\log(1 - \hat{y}^i)$$

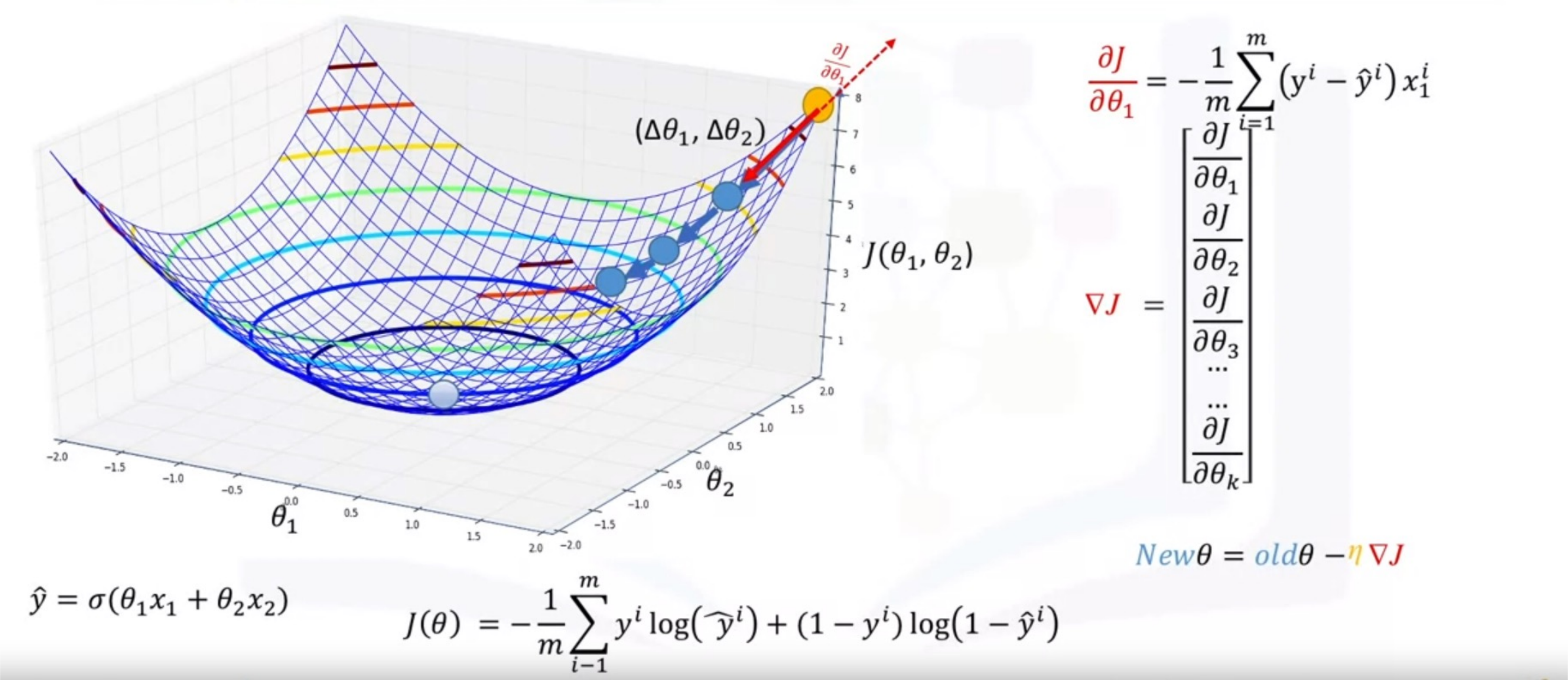
Minimizing loss with gradient descent



$\hat{y} = \sigma(\theta_1 x_1 + \theta_2 x_2)$

$$J(\theta) = -\frac{1}{m}\sum_{i-1}^{m} y^i \log(\hat{y}^i) + (1 - y^i)\log(1 - \hat{y}^i)$$

Minimizing loss with gradient descent



Training steps recap

1. initialize the parameters randomly.
2. Feed the cost function with training set, and calculate the error.
3. Calculate the gradient of cost function.
4. Update weights with new values.
5. Go to step 2 until cost is small enough.
6. Predict the new customer X.

$$\theta^T = [\theta_0, \theta_1, \theta_2, ...]$$

$$J(\theta) = -\frac{1}{m}\sum_{i-1}^{m} y^i \log(\hat{y}^i) + (1 - y^i)\log(1 - \hat{y}^i)$$

$$\nabla J = [\frac{\partial J}{\partial \theta_1}, \frac{\partial J}{\partial \theta_2}, \frac{\partial J}{\partial \theta_3}, \ldots, \frac{\partial J}{\partial \theta_k}]$$

$$\theta_{new} = \theta_{prv} - \eta \nabla J$$

$$P(y=1|x) = \sigma(\theta^T X)$$

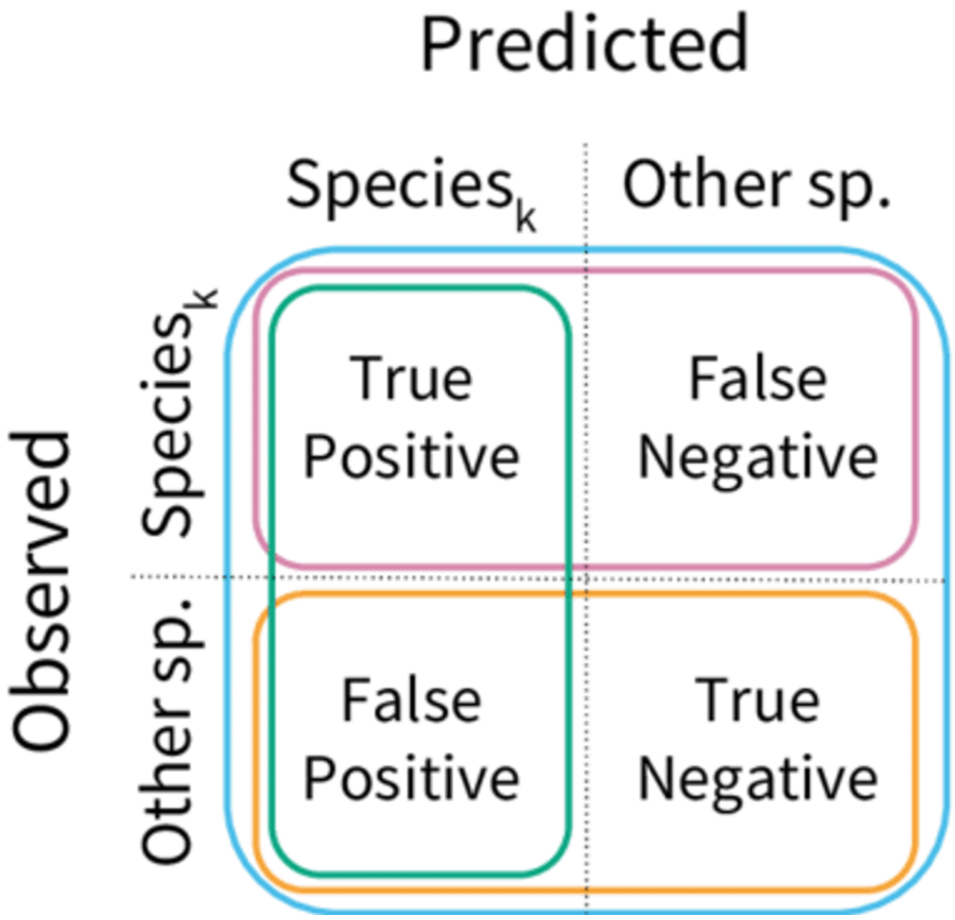
Evaluation Metrics

Individual Number	1	2	3	4	5	6	7	8	9	10	11	12
Actual Classification	1	1	1	1	1	1	1	1	0	0	0	0
Predicted Classification	0	0	1	1	1	1	1	1	1	0	0	0
Result	FN	FN	TP	TP	TP	TP	TP	TP	FP	TN	TN	TN

		Predicted condition	
	Total population = P + N	**Positive (PP)**	**Negative (PN)**
Actual condition	**Positive (P)**	**True positive (TP)**	**False negative (FN)**
	Negative (N)	**False positive (FP)**	**True negative (TN)**

Source: https://en.wikipedia.org/wiki/Confusion_matrix

Evaluation Metrics



Measures the proportion of correctly classified samples

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Measures how many actual negatives are correctly predicted

$$\text{Specificity} = \frac{TN}{TN + FP}$$

Measures how many predicted positives are actually positive

$$\text{Precision} = \frac{TP}{TP + FP}$$

Measures how many actual positives are correctly predicted

$$\text{Recall} = \frac{TP}{TP + FN}$$

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

K-nearest Neighbor

K-NN

X: Independent variable

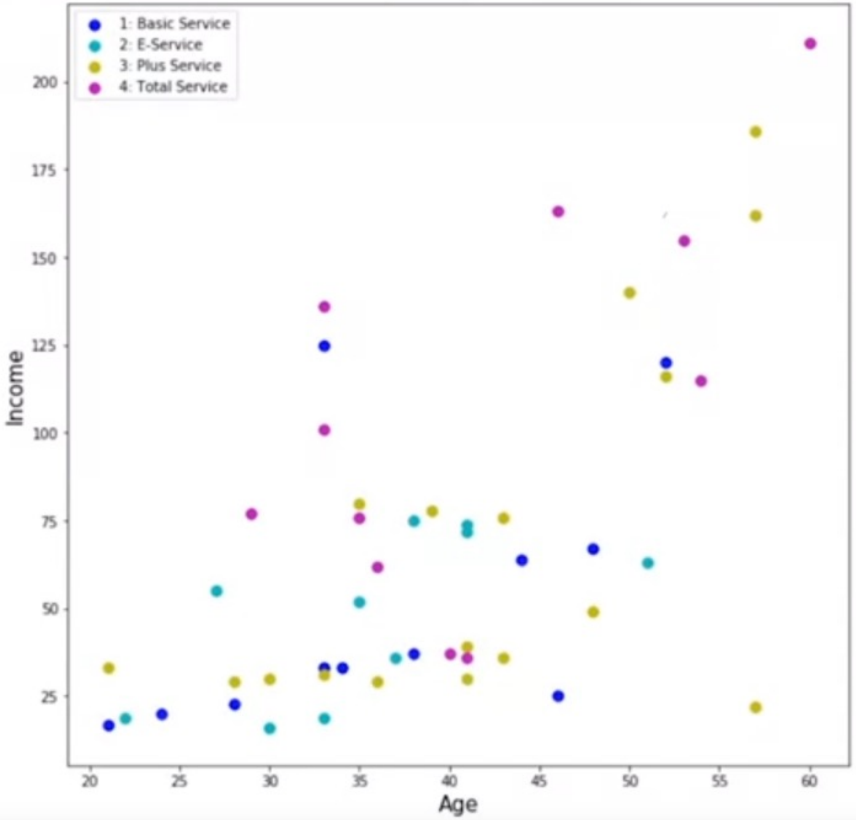
Y: Dependent variable

	region	age	marital	address	income	ed	employ	retire	gender	reside	custcat
0	2	44	1	9	64	4	5	0	0	2	1
1	3	33	1	7	136	5	5	0	0	6	4
2	3	52	1	24	116	1	29	0	1	2	3
3	2	33	0	12	33	2	0	0	1	1	1
4	2	30	1	9	30	1	2	0	0	4	3
5	2	39	0	17	78	2	16	0	1	1	3
6	3	22	1	2	19	2	4	0	1	5	2
7	2	35	0	5	76	2	10	0	0	3	4
8	3	50	1	7	166	4	31	0	0	5	?

Value	Label
1	Basic Service
2	E-Service
3	Plus Service
4	Total Service

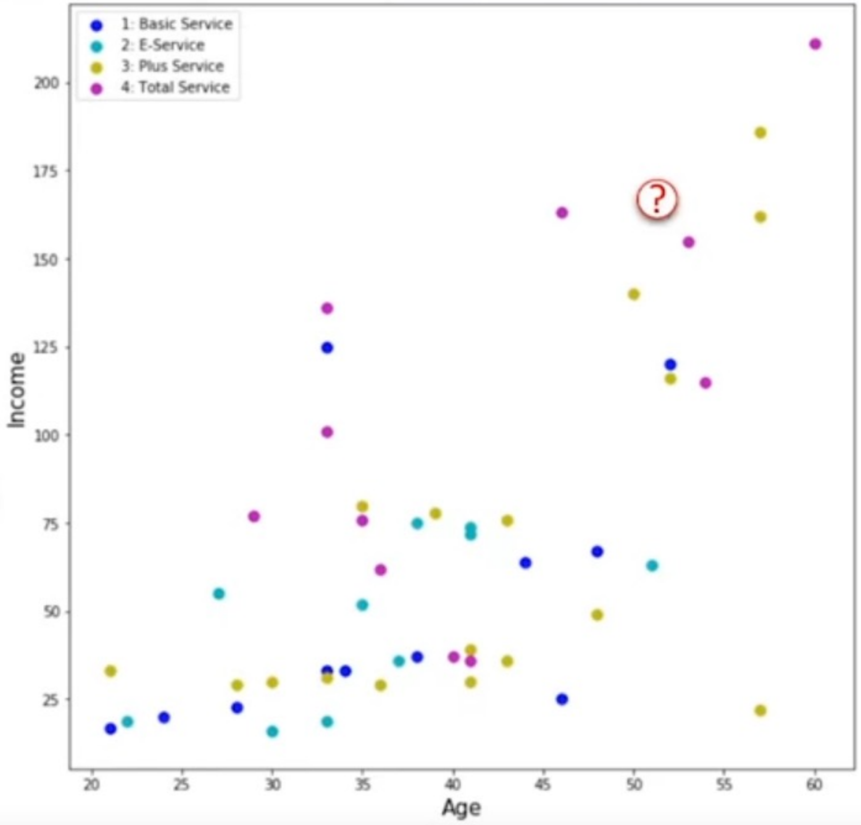
K-NN

	region	age	marital	address	income	ed	employ	retire	gender	reside	custcat
0	2	44	1	9	64	4	5	0	0	2	1
1	3	33	1	7	136	5	5	0	0	6	4
2	3	52	1	24	116	1	29	0	1	2	3
3	2	33	0	12	33	2	0	0	1	1	1
4	2	30	1	9	30	1	2	0	0	4	3
5	2	39	0	17	78	2	16	0	1	1	3
6	3	22	1	2	19	2	4	0	1	5	2
7	2	35	0	5	76	2	10	0	0	3	4
8	3	50	1	7	166	4	31	0	0	5	?



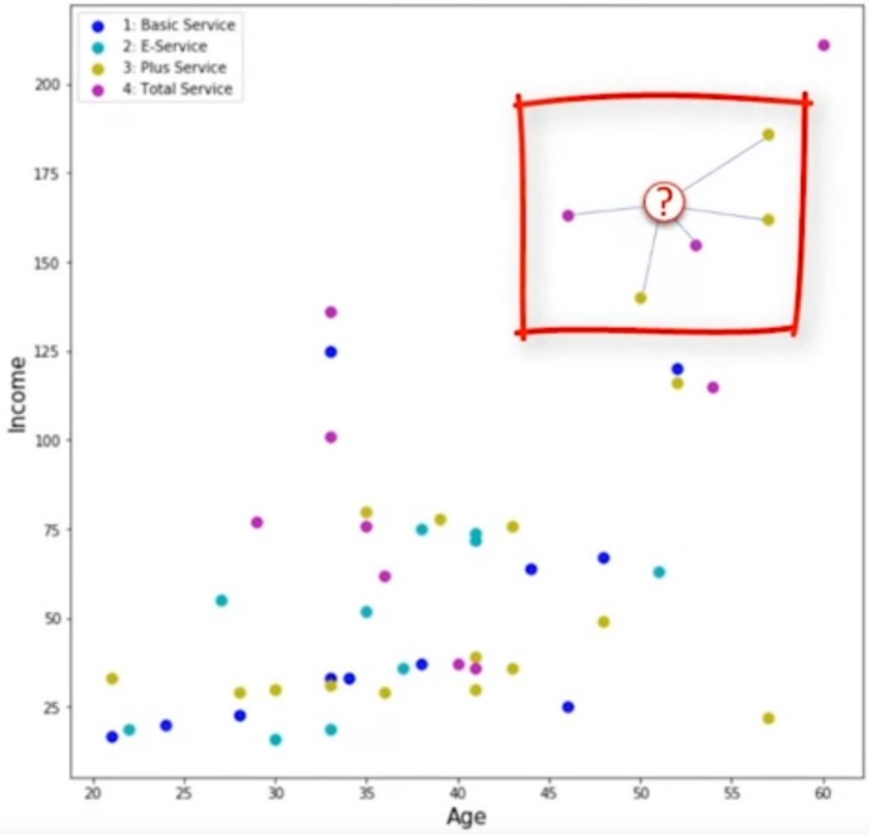
K-NN

	region	age	marital	address	income	ed	employ	retire	gender	reside	custcat
0	2	44	1	9	64	4	5	0	0	2	1
1	3	33	1	7	136	5	5	0	0	6	4
2	3	52	1	24	116	1	29	0	1	2	3
3	2	33	0	12	33	2	0	0	1	1	1
4	2	30	1	9	30	1	2	0	0	4	3
5	2	39	0	17	78	2	16	0	1	1	3
6	3	22	1	2	19	2	4	0	1	5	2
7	2	35	0	5	76	2	10	0	0	3	4
8	3	50	1	7	166	4	31	0	0	5	?



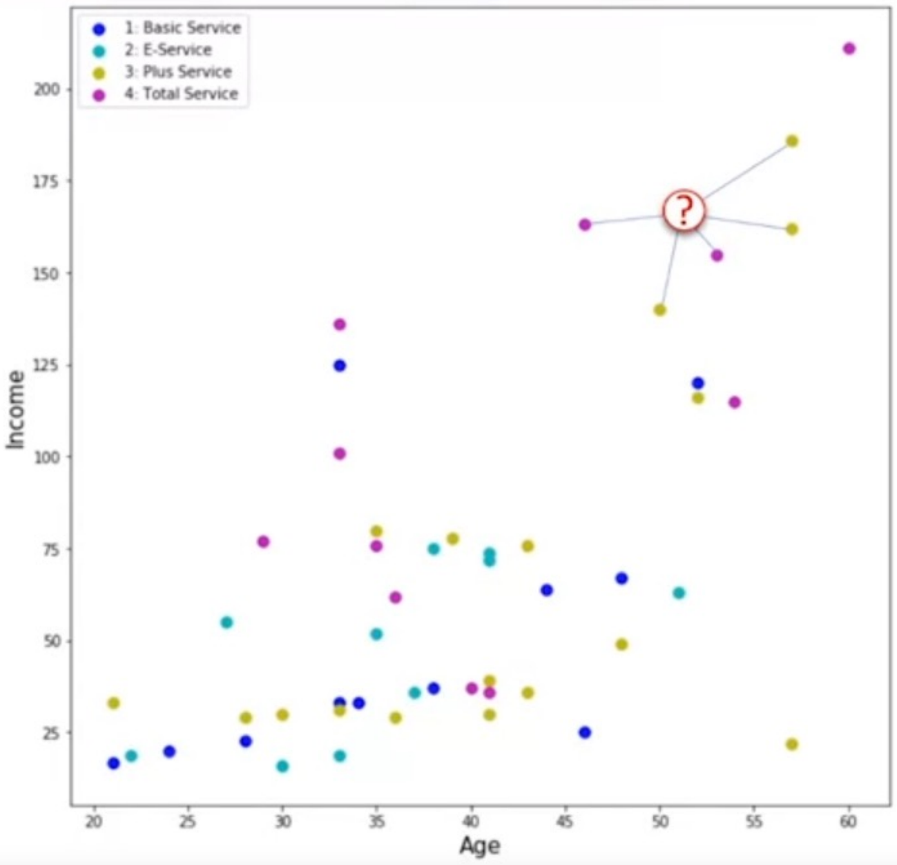
K-NN

	region	age	marital	address	income	ed	employ	retire	gender	reside	custcat
0	2	44	1	9	64	4	5	0	0	2	1
1	3	33	1	7	136	5	5	0	0	6	4
2	3	52	1	24	116	1	29	0	1	2	3
3	2	33	0	12	33	2	0	0	1	1	1
4	2	30	1	9	30	1	2	0	0	4	3
5	2	39	0	17	78	2	16	0	1	1	3
6	3	22	1	2	19	2	4	0	1	5	2
7	2	35	0	5	76	2	10	0	0	3	4
8	3	50	1	7	166	4	31	0	0	5	?



K-NN

- A method for classifying cases based on their similarity to other cases
- Cases that are near each other are said to be "neighbors"
- Based on similar cases with same class labels are near each other



K-NN

1. Pick a value for K.
2. Calculate the distance of unknown case from all cases.
3. Select the K-observations in the training data that are "nearest" to the unknown data point.
4. Predict the response of the unknown data point using the most popular response value from the K-nearest neighbors.

K-NN – Calculate distance

Customer 1
Age
34

Customer 2
Age
30

$$\text{Dis}\,(x_1, x_2) = \sqrt{\sum_{i=0}^{n}(x_{1i} - x_{2i})^2}$$

$$\text{Dis}\,(x_1, x_2) = \sqrt{(34 - 30)^2} = 4$$

K-NN – Calculate distance

Customer 1		
Age	Income	Education
34	190	3

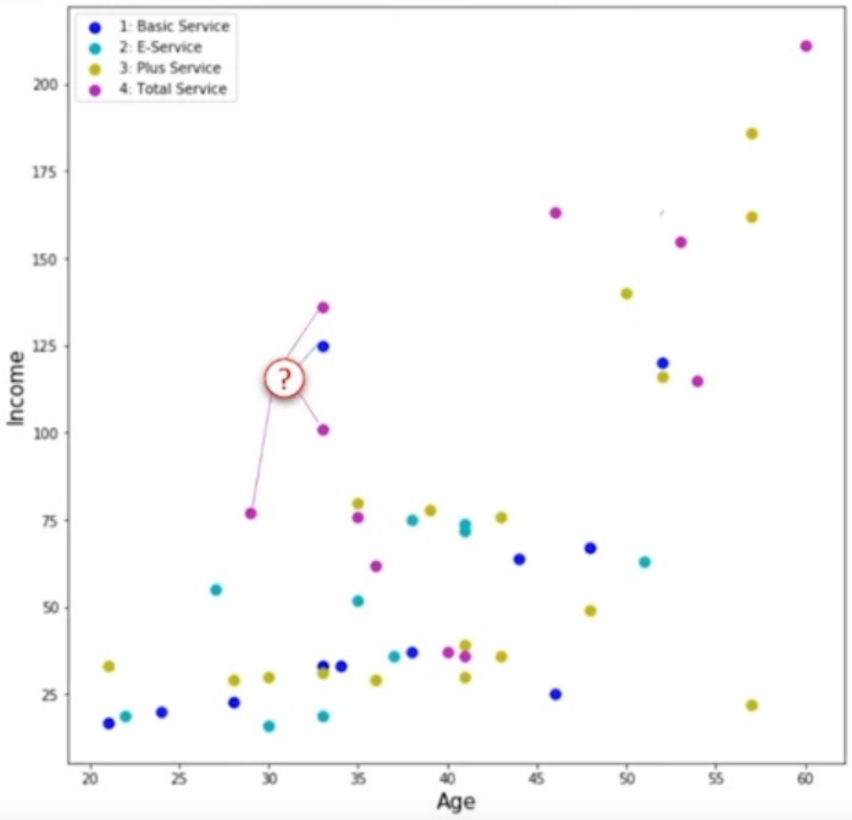
Customer 2		
Age	Income	Education
30	200	8

$$\text{Dis}\ (x_1, x_2) = \sqrt{\sum_{i=0}^{n}(x_{1i} - x_{2i})^2}$$

$$= \sqrt{(34-30)^2+(190-200)^2+(3-8)^2} = 11.87$$

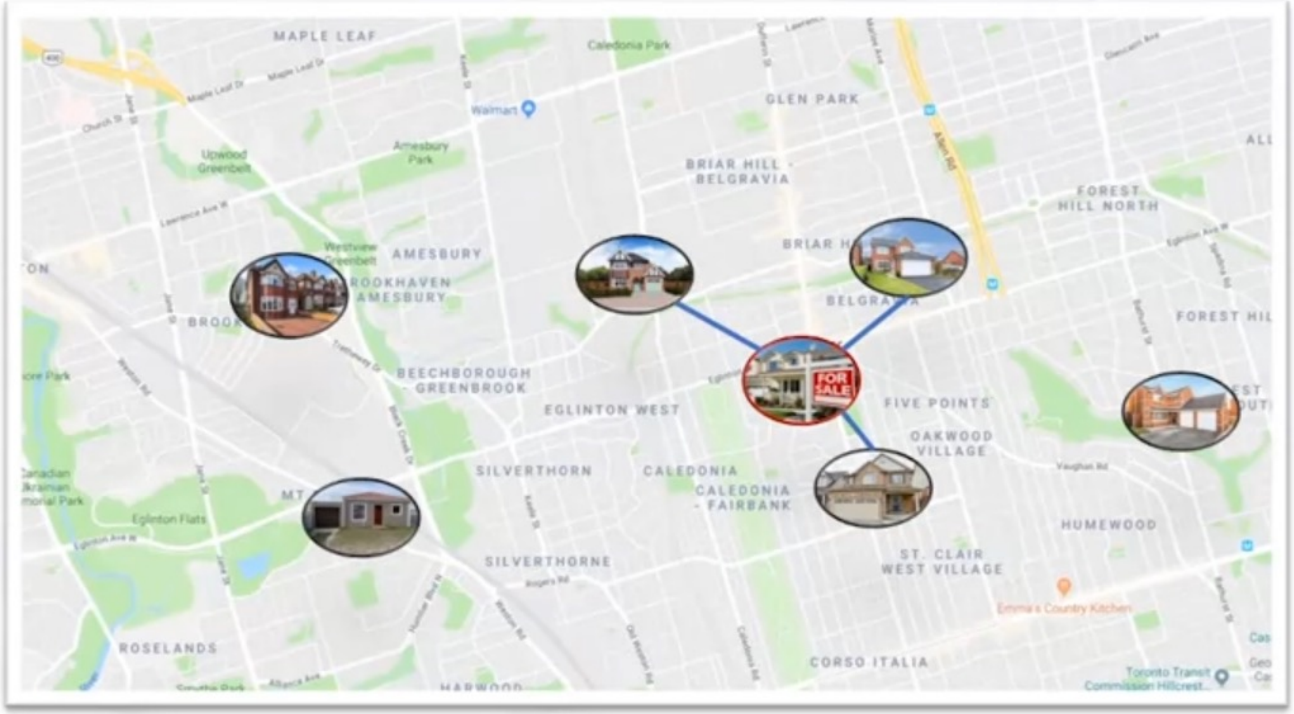
K-NN – Selecting K

- K =1 class 1
- K =20 ?



K-NN

- KNN can also be used for regression



Graded Assignment

- TBA
- Deadline: **Thursday, 04.12.2024**

References

- https://www.coursera.org/learn/machine-learning/
- https://www.coursera.org/learn/machine-learning-with-python/
- https://www.ejable.com/tech-corner/ai-machine-learning-and-deep-learning/logistic-and-linear-regression/#Logistic_regression
- https://www.youtube.com/watch?v=yIYKR4sgzI8
- https://www.visual-design.net/post/top-machine-learning-algorithms-classification
- https://en.wikipedia.org/wiki/Confusion_matrix