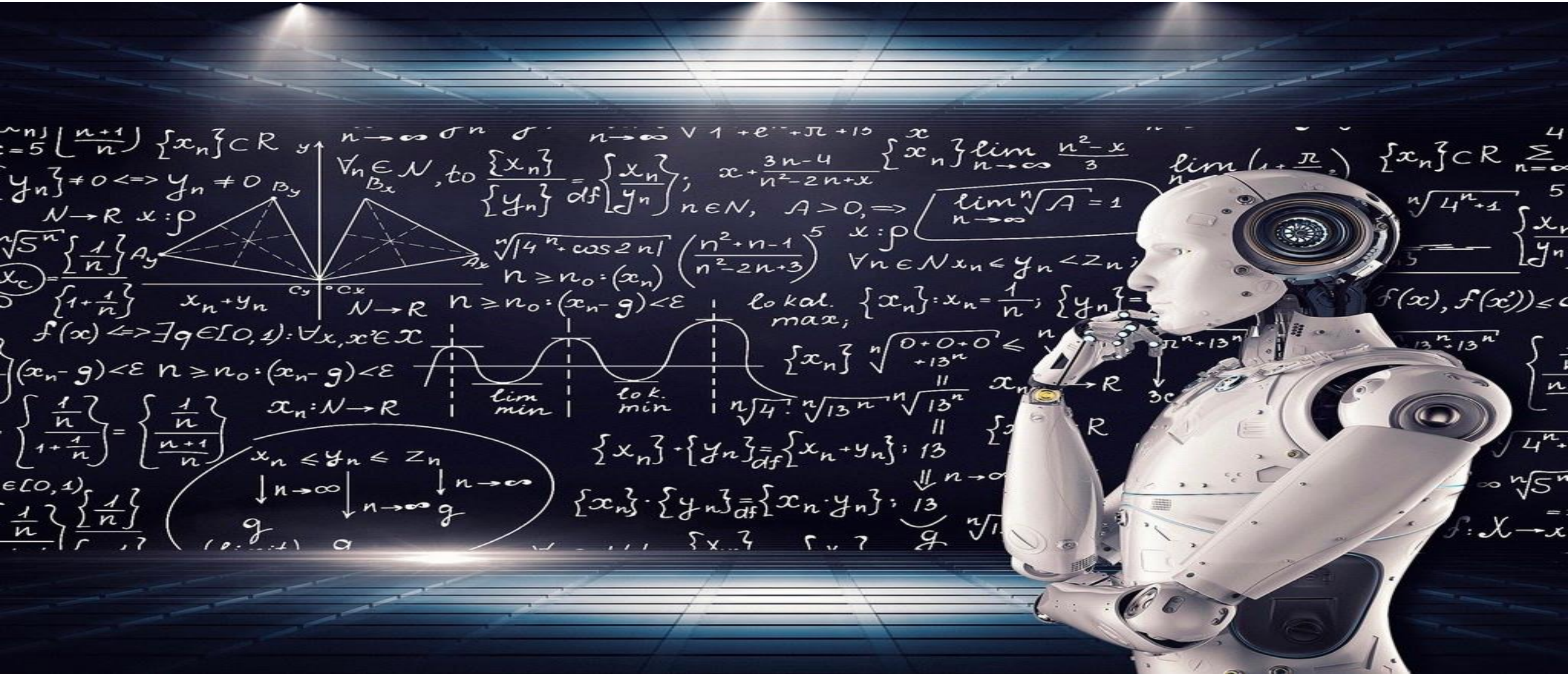
Final Recitation



Linear Regression

- Let $D = \left(\{x^{(1)}, y^{(1)}\}, \{x^{(2)}, y^{(2)}\}, \dots, \{x^{(m)}, y^{(m)}\}\right)$ denote our training set where $x^{(i)} \in \mathbb{R}^d$ and $y \in \mathbb{R}$

- We are looking for $\underset{\theta_0,\theta}{\operatorname{argmin}} J(\theta_0, \theta)$, $J(\theta_0, \theta) = \frac{1}{2m} \sum_{i=1}^{m} \left((\theta^T x^{(i)} + \theta_0) - y^{(i)}\right)^2$

  and $\theta_0 \in \mathbb{R}, \theta \in \mathbb{R}^d$.

*for simplicity we will use only θ to denote our parameters in the next slides.

Linear Regression – Question 1

- What are the 2 techniques we showed for solving linear regression?

- Gradient Descent ---- Analytically using pinv

Guess some value for θ

Repeat until error is small enough

Update $\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta)$

// α = learning rate

Solve $\left|\left|X\theta - y\right|\right|^2$

Solution : $\theta = pinv(X)y$

$$pinv(X) = (X^T X)^{-1} X^T$$

Linear Regression – Question 1

- What are the 2 techniques we showed for solving linear regression?
- Gradient Descent ---- Analytically using pinv

- Give an argument for preferring pinv
  - Guaranteed to find global minimum
- Give an argument for preferring GD.
  - Computationally in-feasible to solve analytically for very large datasets

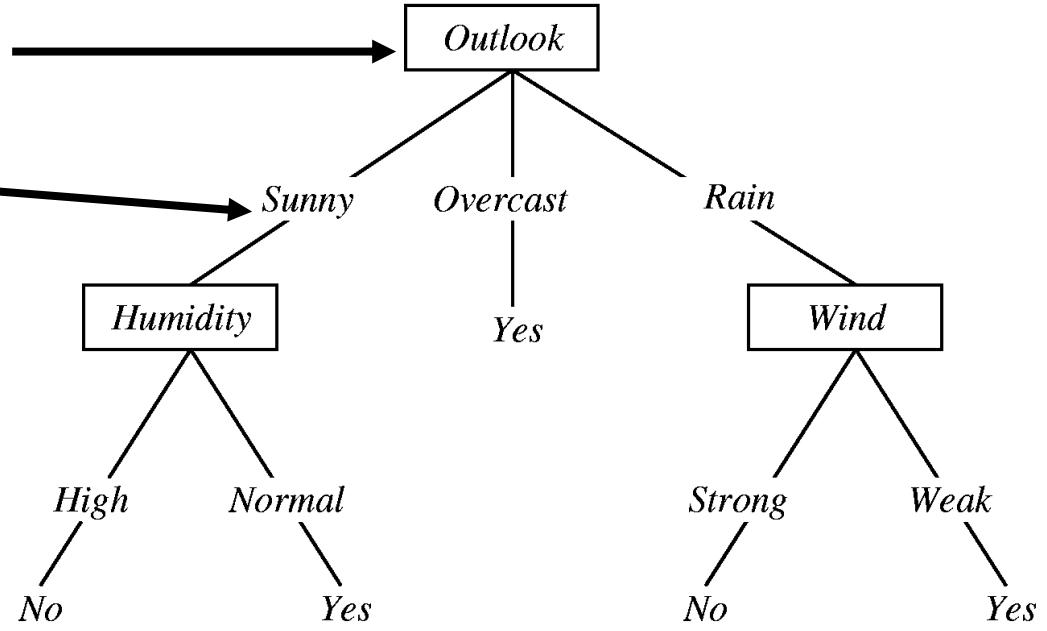
Linear Regression – Question 2

- For the rest of the question, assume we decided to go with Gradient Descent
- Can we change the error function to be:
  - $J(\theta) = \frac{1}{2m} \sum_{i=1}^{m} \left((\theta^T x^{(i)}) - y^{(i)} \right)$
  - No! This can cause the error to be negative and so minimizing this value can simply go to $-\infty$
- Can we change the error function to be:
  - $J(\theta) = \frac{1}{2m} \sum_{i=1}^{m} \left| \left((\theta^T x^{(i)}) - y^{(i)} \right) \right|$
  - Yes, But we will need to change the way we compute the gradients which will be slightly trickier.

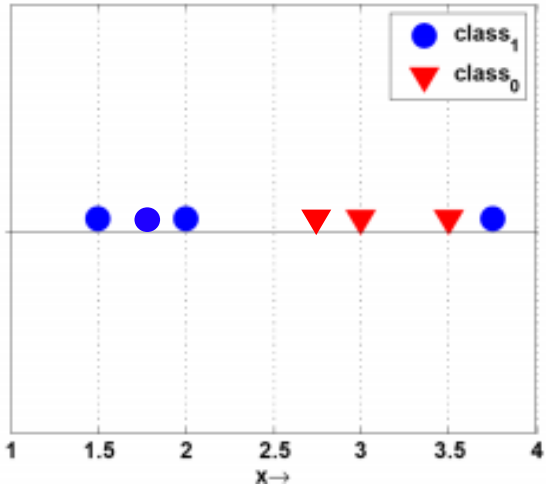
Decision Trees

- Let $D = \left(\{x^{(1)}, y^{(1)}\}, \{x^{(2)}, y^{(2)}\}, \dots, \{x^{(m)}, y^{(m)}\}\right)$ denote our training set where $x^{(i)} \in \mathbb{R}^d$ and $y \in \{1, \dots, c\}$.

- Tree Structure:
  - Each internal node (all nodes except the leaves) tests an attribute
  - Each branch corresponds to attribute value
  - Each leaf node assigns a classification



Decision Trees – Question 1



- What is the training set error of DT1 (only 1 Boolean split) on the data?
  - 1
- What is its leave one out cross validation error count?
  - 1
- And with a DT* (Split till perfect classification)?
- Training set?
  - 0
- Leave one out cross validation error count?
  - 2

Decision Trees – Question 2

- You've just finished training a decision tree, and it is getting abnormally bad performance on both your **training** and **test** sets. You know that your implementation has no bugs, so what could be causing the problem?

1. You haven't used chi pruning.
2. Your tree is too shallow.
3. Your learning rate is too high.
4. You made a bad train-test split.

Decision Trees – Question 2

- You’ve just finished training a decision tree, and it is getting abnormally bad performance on both your **training** and **test** sets. You know that your implementation has no bugs, so what could be causing the problem?

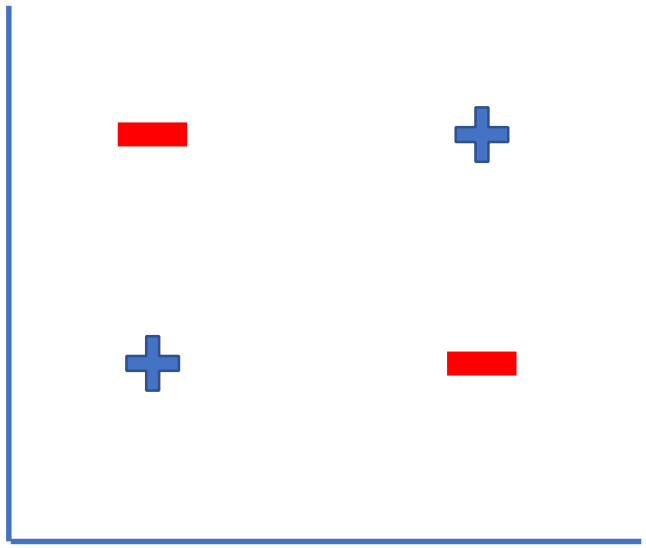
1. You haven’t used chi pruning.
2. Your tree is too shallow.
3. Your learning rate is too high.
4. You made a bad train-test split.

Decision Trees – Question 3

- Give an example with 2 features where a DT1 will have 50% accuracy on the data, while a DT2 will have 100% accuracy.

Decision Trees – Question 3

- Give an example with 2 features where a DT1 will have 50% accuracy on the data, while a DT2 will have 100% accuracy.



SVM

- Let $D = \left(\{x^{(1)}, y^{(1)}\}, \{x^{(2)}, y^{(2)}\}, \dots, \{x^{(m)}, y^{(m)}\}\right)$ denote our training set where $x^{(i)} \in \mathbb{R}^d$ and $y \in \{-1, 1\}$.

  Our goal:

- Minimize

$$\frac{1}{2} \cdot \|w\|^2$$

- Subject to:

$$t_d(w^T x_d + w_0) \geq 1$$

SVM – Dual form

- Maximize

$$\sum_{d} \alpha_d - 1/2 \sum_{d} \sum_{e} \alpha_d \alpha_e t_d t_e x_d^T x_e$$

- Subject to:

$$\sum_{d} \alpha_d t_d = 0$$

$$\alpha_d \geq 0$$

SVM – Soft margin

- Minimize

$$\frac{1}{2}\|w\|^2 + \gamma \sum_d \xi_d$$

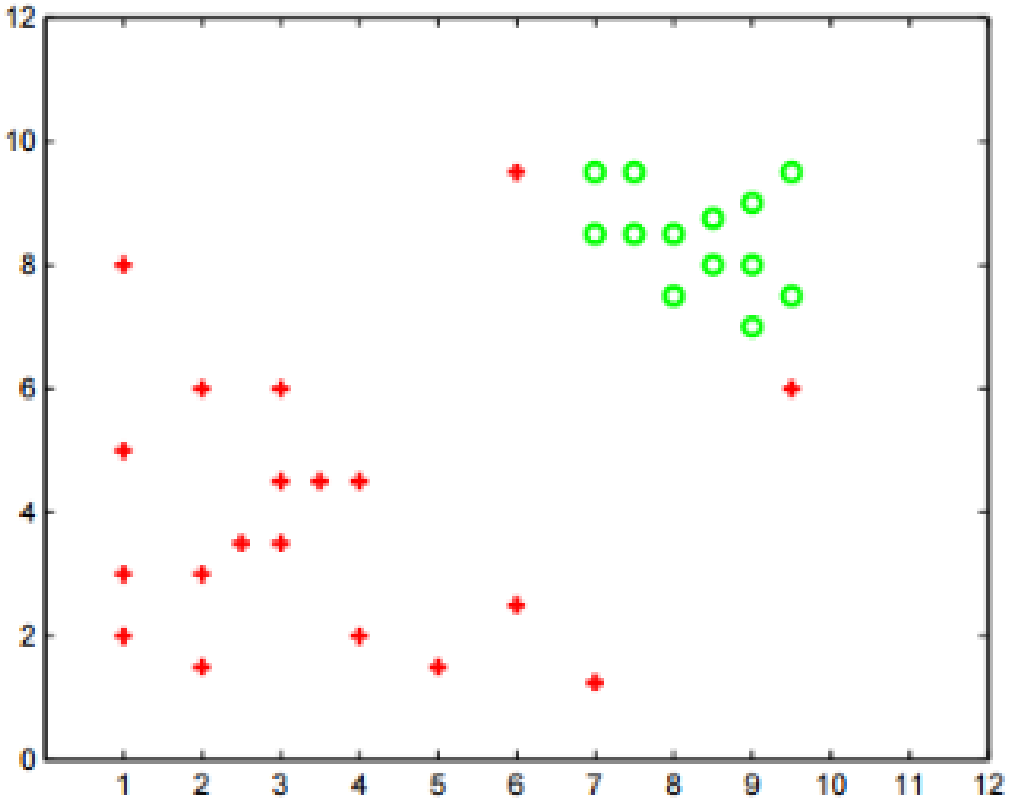
- Subject to:

$$t_d(w^T x_d + w_0) \geq 1 - \xi_d \quad \xi_d \geq 0$$

SVM – Q1

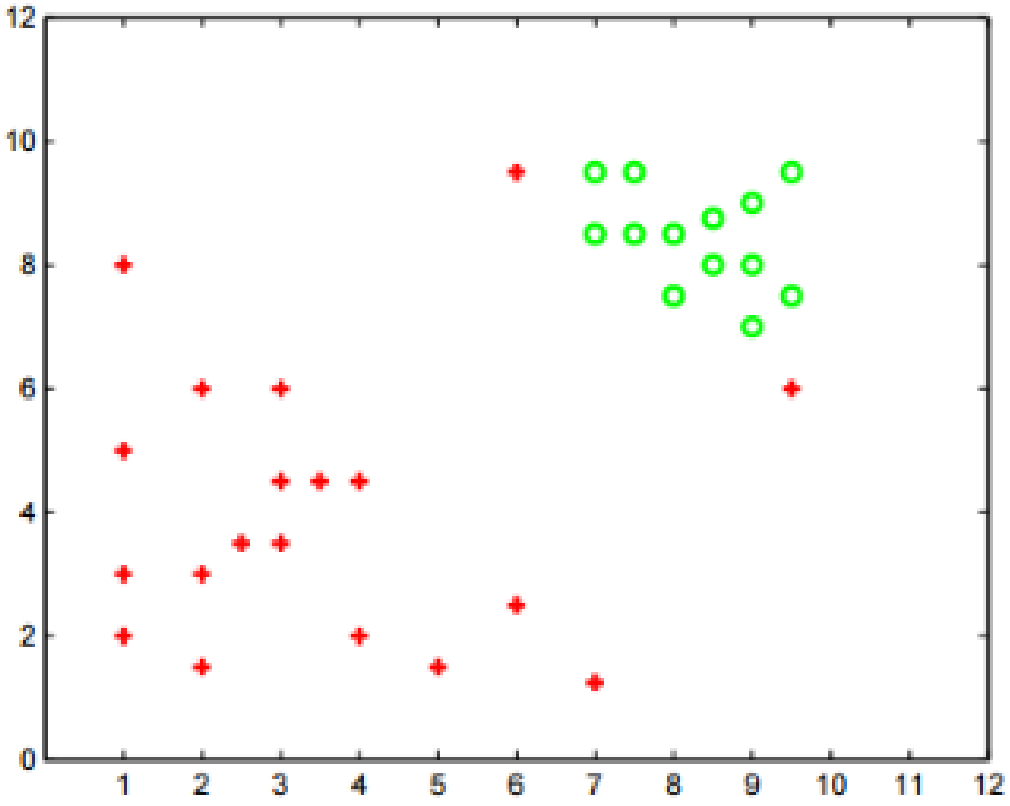
- For the following problem, assume that we are training an SVM with a quadratic kernel– that is, our kernel function is a polynomial kernel of degree 2. You are given the data set presented in the figure.

  The slack penalty γ will determine the location of the separating hyperplane.



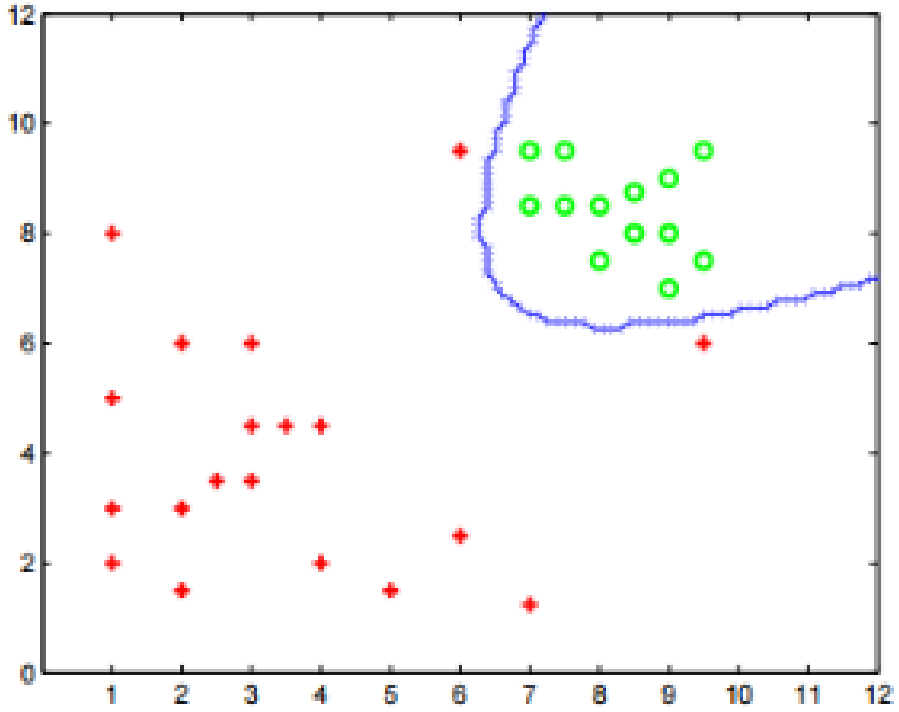
SVM – Q1

- Where would the decision boundary be for very large values of γ (i.e., $\gamma \to \infty$)?



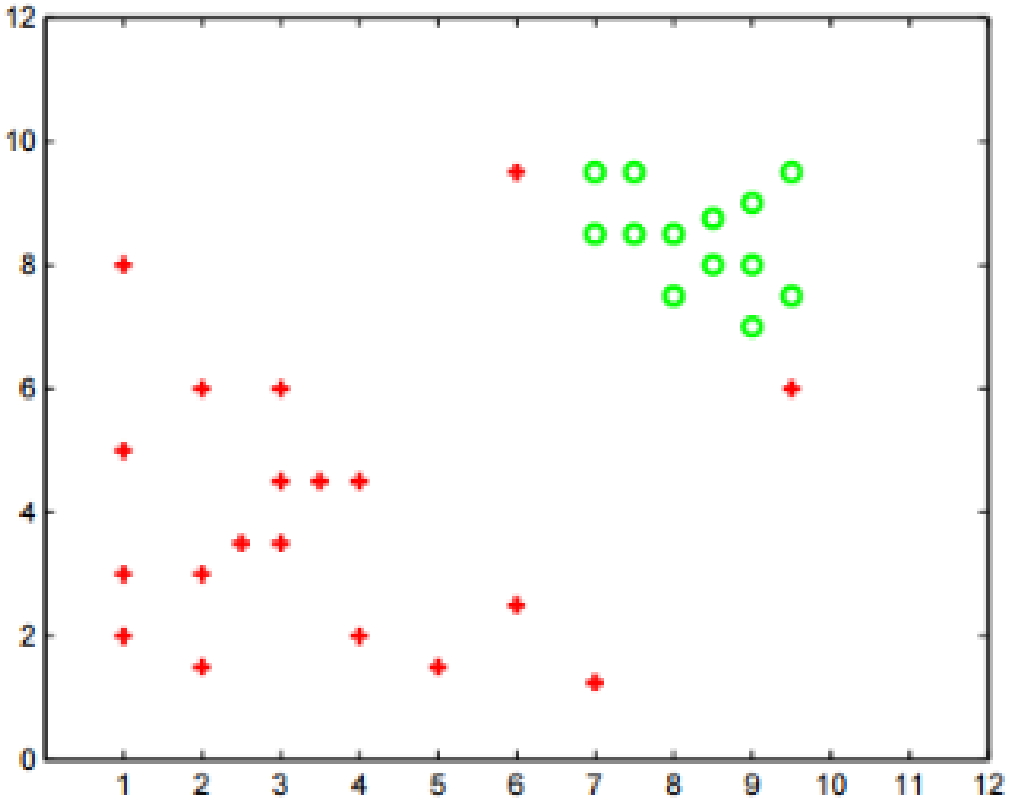
SVM – Q1

- Where would the decision boundary be for very large values of γ (i.e., $\gamma \to \infty$)?



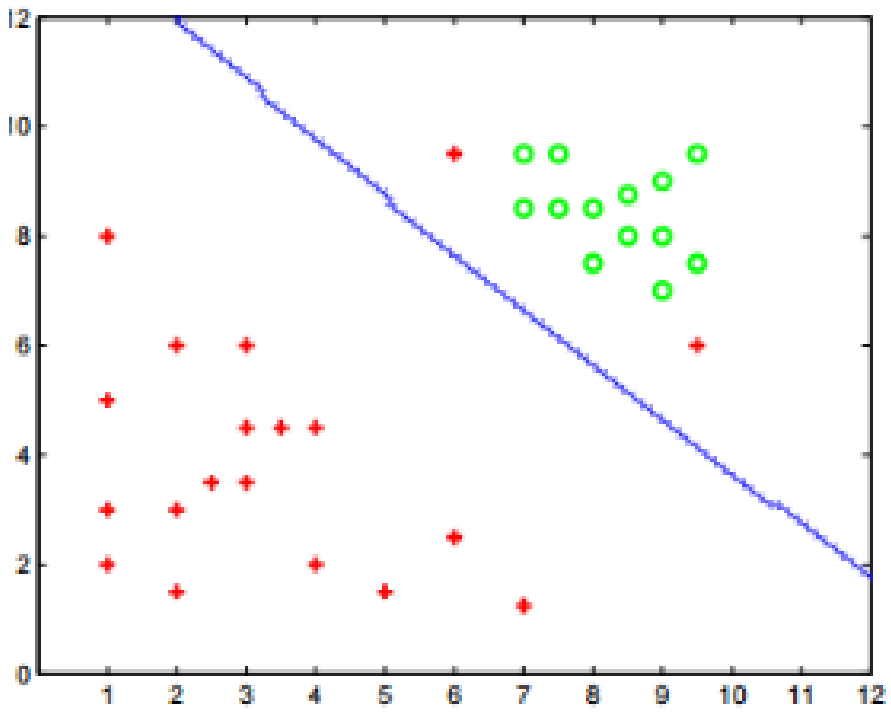
SVM – Q1

- Where would the decision boundary be for very small values of γ (i.e., $\gamma \to 0$)?



SVM – Q1

- Where would the decision boundary be for very small values of γ (i.e., $\gamma \to 0$)?

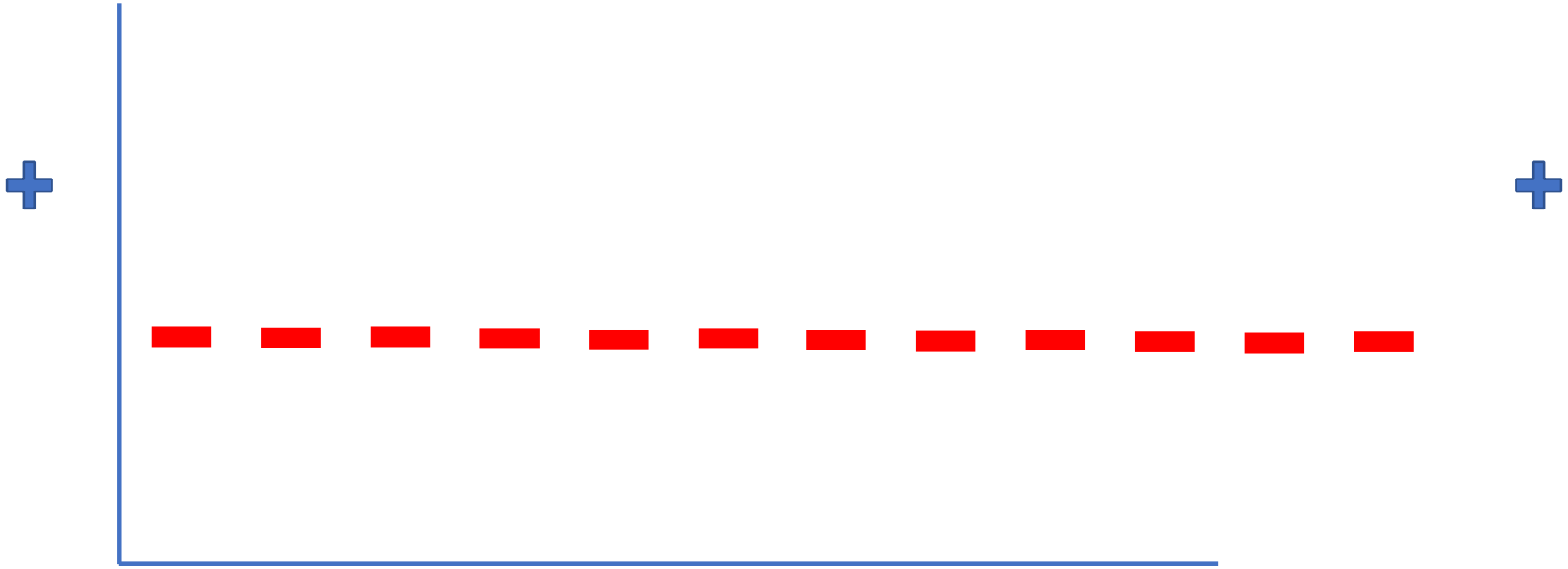


SVM – Q2

- Suppose we train a hard-margin linear SVM on $n > 100$ data points in $\mathbb{R}^2$, yielding a hyperplane with exactly 2 support vectors. If we add one more data point and retrain the classifier, what is the **maximum** possible number of support vectors for the new hyperplane (assuming the n + 1 points are linearly separable)?

SVM – Q2

- what is the **maximum** possible number of support vectors for the new hyperplane (assuming the n + 1 points are linearly separable)?
- $n + 1$



VC Dimension

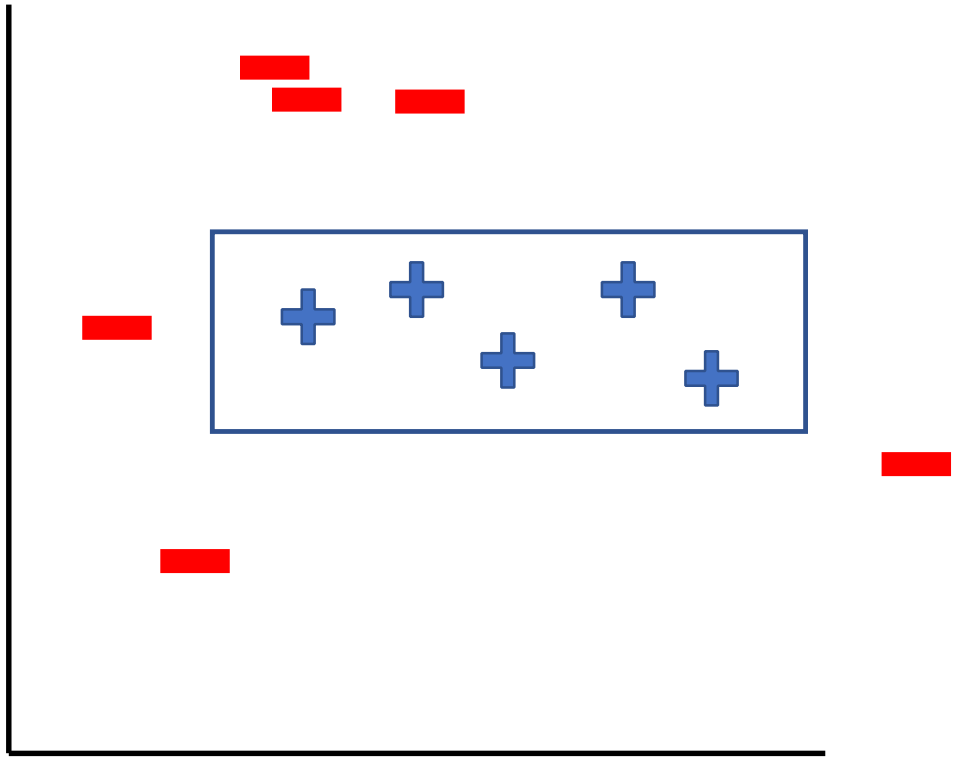
- *VC(H)*, VC dimension of *H*, defined over instance space U, is the size of the largest finite subset of *U* **shattered** by hypothesis space *H*

- An hypothesis class H **shatters** a set of points $\mathrm{X} = \{x_1, x_2, \ldots, x_m\} \in U$ iff for **<u>every</u>** assignment $\mathrm{Y} = \{y_1, y_2, \ldots, y_m\} \in \{-1, 1\}^m$, there exists $h \in H$ s.t $\forall i\colon\ h\,(x_i) = y_i$

VC Dimension – Q1

- Let $U = \mathbb{R}^2$ and H be the set of axis aligned rectangles, s.t. all points inside the rectangle are label +.

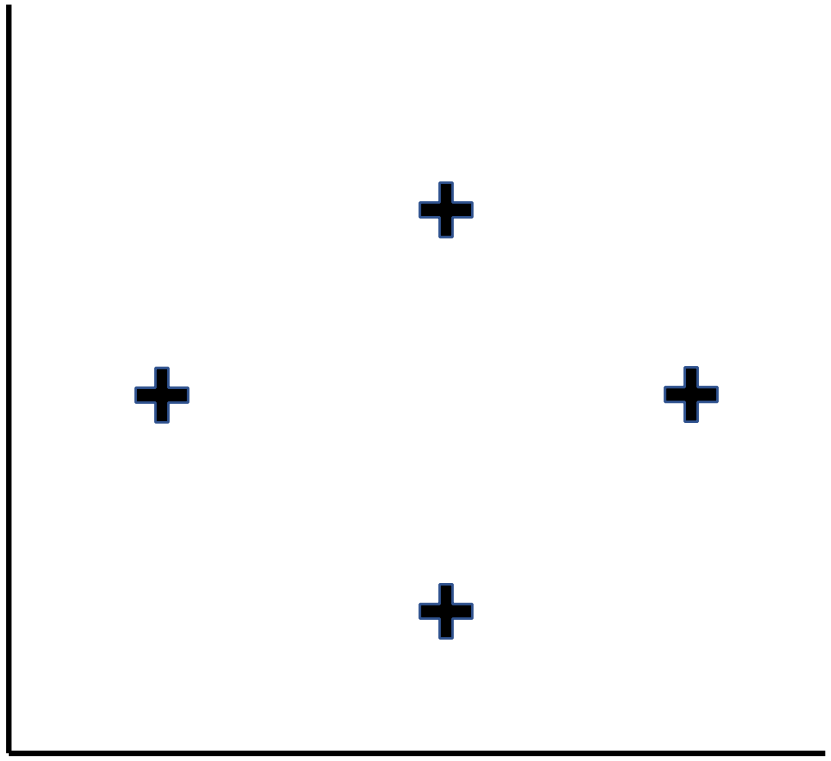
  What is the VC Dimension of H?



VC Dimension – Q1 - Solution

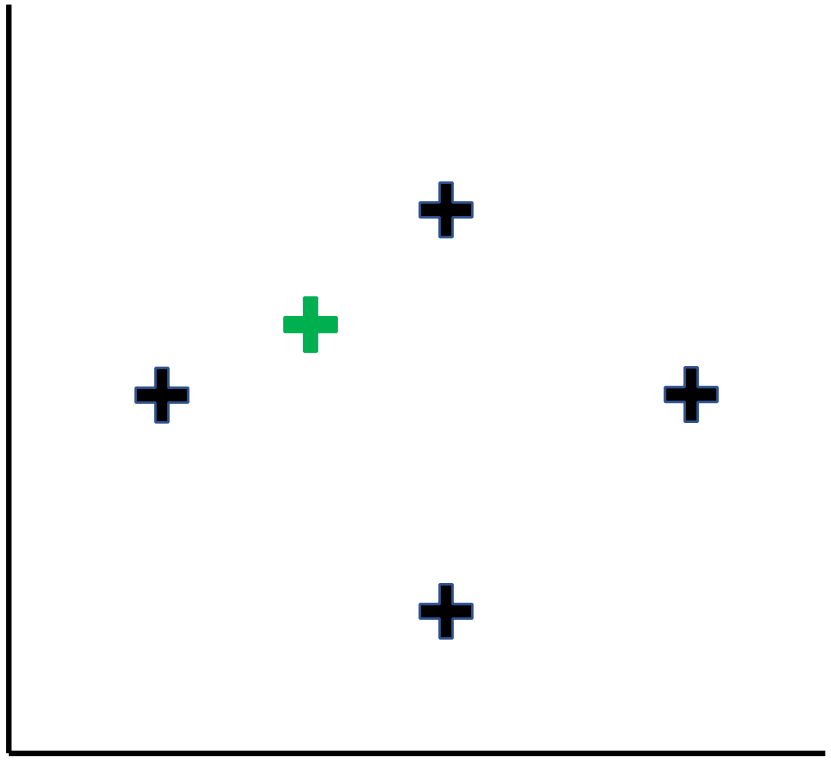
- First let's show that $VC(H) \geq 4$
- Easy to see that we have an $h \in H$

  For each labeling.



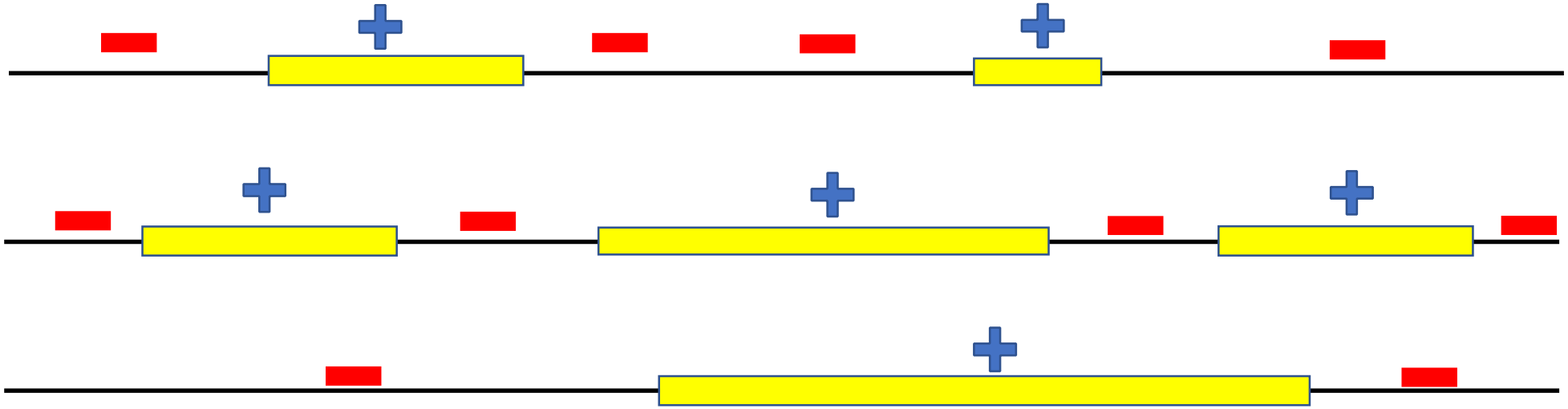
VC Dimension – Q1 - Solution

- Now, let's show that $VC(H) > 5$
- For any set of 5 points there must be some point that is "internal", i.e., is neither the extreme left, right, top or bottom point of the five.
  If we label this internal point as negative and the remaining 4 points as positive then there is no axes-aligned rectangle which could realize this labeling



VC Dimension – Q2 - Solution

- Let $U = \mathbb{R}$ and H be a finite union of intervals on the line.
- What is the VC dimension of H?



VC Dimension – Q2 - Solution

- Let $U = \mathbb{R}$ and H be a finite union of intervals on the line.
- What is the VC dimension of H?
  - $VC(H) = \infty$
  - Why? For every m we can pick the set $\mathrm{X} = \{1, 2, 3, \dots, m\}$ then for every labeling l we simply pick the hypothesis : $h = \{(a_i - 0.1, a_i + 0.1): a_i \in X \wedge l(a_i)=1\}$

Knn

- Let $D = \left(\{x^{(1)}, y^{(1)}\}, \{x^{(2)}, y^{(2)}\}, \dots, \{x^{(m)}, y^{(m)}\}\right)$ denote our training set where $x^{(i)} \in \mathbb{R}^d$ and $y \in \{1, \dots, c\}$ or $y \in \mathbb{R}$.

- Instance based learning. Prediction is based on k nearest neighbors.

  - For regression:

$$\hat{f}(x) = \frac{1}{k}\sum_{i=1}^{k} y^{(i)}$$

  - For classification:

$$\hat{f}(x) = MAJ_i\left(\{y^{(i)}\}\right)$$

Knn – Question 1

x	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
y	A	A	A	A	B	A	A	A	A	B	B	B	B	A	B	B	B	B

- What would be the classification of a test sample with x = 4.2 according to 1-NN ?
  - B
- What would be the classification of a test sample with x = 4.2 according to 3-NN ?
  - A
- What is the “leave-one-out” cross validation error of 1-NN. If you need to choose between two or more examples of identical distance, make your choode so that the number of errors is maximized.
  - 8
- What is the “leave-one-out” cross validation error of 17-NN. If you need to choose between two or more examples of identical distance, make your choice so that the number of errors is maximized.
  - 18

K - means

- Let $D = \{x^{(1)}, x^{(2)}, \dots x^{(m)}\}$ denote our training set where $x^{(i)} \in \mathbb{R}^d$

- We are interested in finding disjoint clusters $A_1, A_2, \dots, A_k$ $(\cup A_i = D)$ and center points $\mu_1, \dots, \mu_k$ which will minimize our objective function :

$$\sum_{i=1}^{k} \sum_{x \in A_i} \|x - \mu_i\|^2$$

K – means – Question 1

- Consider the following modification to the k-means algorithm :
  Instead of choosing the mean value of the points as the new center choose the data point in the cluster which minimizes the object function inside the cluster. This is known as the k-mediods.
- You argue with a co-worker that the k-means algorithm is better since it will surely produce a better solution for each cluster than the k-mediods. You co-worker say that choosing the mean value is intuitively correct but does not necessarily lead to the optimal solution in a cluster.

  How would you prove him wrong?

K – means – Question 1

- All you need to do is derive the objective function and show that the optimal solution is given at the mean value.

  For simplicity we show the case for 1 dimension

- Focusing on a single cluster :
- $\frac{\mathrm{d}f}{du}\sum_{x\in D_i}\|x-\mu_i\|^2 = \frac{\mathrm{d}f}{du}[\left(\sqrt{(x^1-\mu_i)^2}\right)^2 + \ldots + \left(\sqrt{(x^m-\mu_i)^2}\right)^2] =$
  $\frac{\mathrm{d}f}{du}[(x^1-\mu_i)^2 + \ldots + (x^m-\mu_i)^2] =$
  $-2(x^1-\mu_i) - 2(x^1-\mu_i) - \cdots - 2(x^m-\mu_i) =$
  $-2(x^1+\ldots+x^m) + 2m\mu_i.$

- Setting to zero yields :

$$\mu_i = \frac{x^1+\ldots+x^m}{m}$$

Bayes Classifier

- When will Full Bayes get better results compared to Naive Bayes? Explain

- Naïve Bayes assumption is that the features are independent given the class
- Full Bayes can achieve better results only when there is some dependency between the features given the class
- But, this is not enough. Full Bayes will get better results only when ignoring this dependency will change the prediction = Naïve Bayes will get max posterior for different class

* Obviously, most of the time Full Bayes is not practical and therefore we will use only Naïve Bayes

MAP Classifier

- Let $D = \left(\{x^{(1)}, y^{(1)}\}, \{x^{(2)}, y^{(2)}\}, \dots, \{x^{(m)}, y^{(m)}\}\right)$ denote our training set where $x^{(i)} \in \mathbb{R}^d$ and $y \in \{1, \dots, c\}$ or $y \in \mathbb{R}$.

- $Prediction: \underset{i}{argmax}\, P(A_i|x) = \frac{P(x|A_i) \cdot P(A_i)}{P(x)}$

- $P(A_i)\ - prior$
- $P(x|A_i) - likelihood$
- $P(A_i|x)\ - posterior$

MAP Classifier – Question 1

- Given the following:

Class / $X_1 =$	C_1	C_2
-1	0.2	0.3
0	0.4	0.6
1	0.4	0.1

Class / $X_2 =$	C_1	C_2
-1	0.4	0.1
0	0.5	0.3
1	0.1	0.6

- Assume $P(C_1) = P(C_2) = 0.5$
- Classify the following under the Naïve assumption : (-1, 1)

MAP Classifier – Question 1

- Classify the following under the Naïve assumption : (-1, 1)

$X_1 =$ \ Class	C_1	C_2
-1	0.2	0.3
0	0.4	0.6
1	0.4	0.1

$X_2 =$ \ Class	C_1	C_2
-1	0.4	0.1
0	0.5	0.3
1	0.1	0.6

- $P(-1, 1|C_1) = 0.2 \cdot 0.1 \cdot 0.5 = 0.02$
- $P(-1, 1|C_2) = 0.3 \cdot 0.6 \cdot 0.5 = 0.18$
- So C_2 Will be chosen.

MAP Classifier – Question 1

- Classify the following under the Naïve assumption : (-1, 1)

Class / $X_1 =$	C_1	C_2
-1	0.2	0.3
0	0.4	0.6
1	0.4	0.1

Class / $X_2 =$	C_1	C_2
-1	0.4	0.1
0	0.5	0.3
1	0.1	0.6

- Assuming the table values stays the same, What can we change in order to switch the classification?
  - Change the priors. (What is smallest P(A) which will switch the classification?)
  - Solve $0.2 * 0.1 * x = 0.3 * 0.6 * (1 - x) \rightarrow x = 0.899$
  - So set P(A) = 0.9

General Questions

- Consider a function over n Binary features, defined as follows:
  - If at least k variables are false (0), then the class is **positive**, otherwise the class is **negative**

- Can you represent this function using a linear threshold function?
  - If your answer is **YES**, then give a precise numerical setting of the weights.

  - If **NO**, clearly explain why this function cannot be represented using a linear threshold function.

- Example : $k = 3 \; and \; n = 6$
- $if \; x = [1,1,0,0,0,1] \; x$ is labeled as positive
- $if \; x = [1,0,0,0,0,1] \; x$ is labeled as positive
- $if \; x = [1,1,0,1,0,1] \; x$ is labeled as negative

General Questions

- **YES**. Consider the following setting of weights:
  $w_i = -1$ for all i and $w_0 = \mathrm{n} - \mathrm{k} + 0.5$
- If at least k variables are false (0), then

$$\sum_{i=1} w_i x_i \geq -n + k$$

- Therefore:

$$w_0 + \sum_{i=1} w_i x_i \geq -n + k + n - k + 0.5 \geq 0.5 > 0$$

- On the other hand, if fewer than k variables are false, then

$$\sum_{i=1} w_i x_i \leq -\big(n - (k-1)\big) = -n + k - 1$$

- Therefore:

$$w_0 + \sum_{i=1} w_i x_i \leq -n + k - 1 + n - k + 0.5 \leq -0.5 < 0$$

General Questions

- Let H = a polynomial of any degree, s.t. $h(x) = \begin{cases} 1 & if\ p(x) \geq 0 \\ -1 & otherwise \end{cases}$ and let our instance space be $\mathbb{R}$.

- Given a dataset $\mathrm{D} = \left(\{x^{(1)}, y^{(1)}\}, \{x^{(2)}, y^{(2)}\}, \dots, \{x^{(m)}, y^{(m)}\}\right)$ with binary labels find an hypothesis which return the following:

$$h(x) = \begin{cases} 1 & if\ (x, y) \in D \wedge y = 1 \\ -1 & otherwise \end{cases}$$

- Meaning h return 1 for the instances in our training set and 0 for all the other instances in $\mathbb{R}$

General Questions

- Given a dataset $\mathrm{D} = \left(\{x^{(1)}, y^{(1)}\}, \{x^{(2)}, y^{(2)}\}, \dots, \{x^{(m)}, y^{(m)}\}\right)$ with binary labels

  find an hypothesis which return the following:

$$h(x) = \begin{cases} 1 & if\ (x, y) \in D \wedge y = 1 \\ -1 & otherwise \end{cases}$$

- Let $I = \{i : y^{(i)} = 1\}$ be the index set of all positive instances.
  Set

$$p(x) = -\left(\prod_{i \in I} \left(x - x^{(i)}\right)\right)^2$$