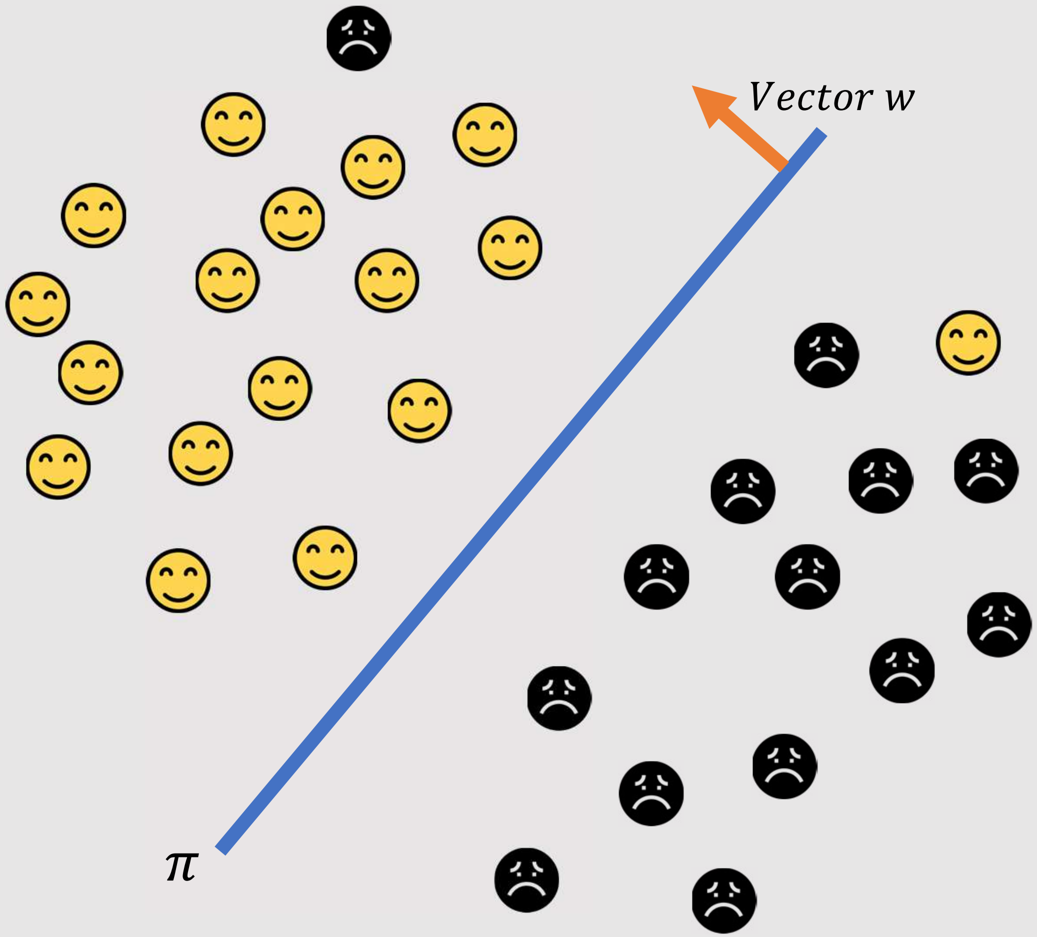
Logistic Regression

Geometric Intuition

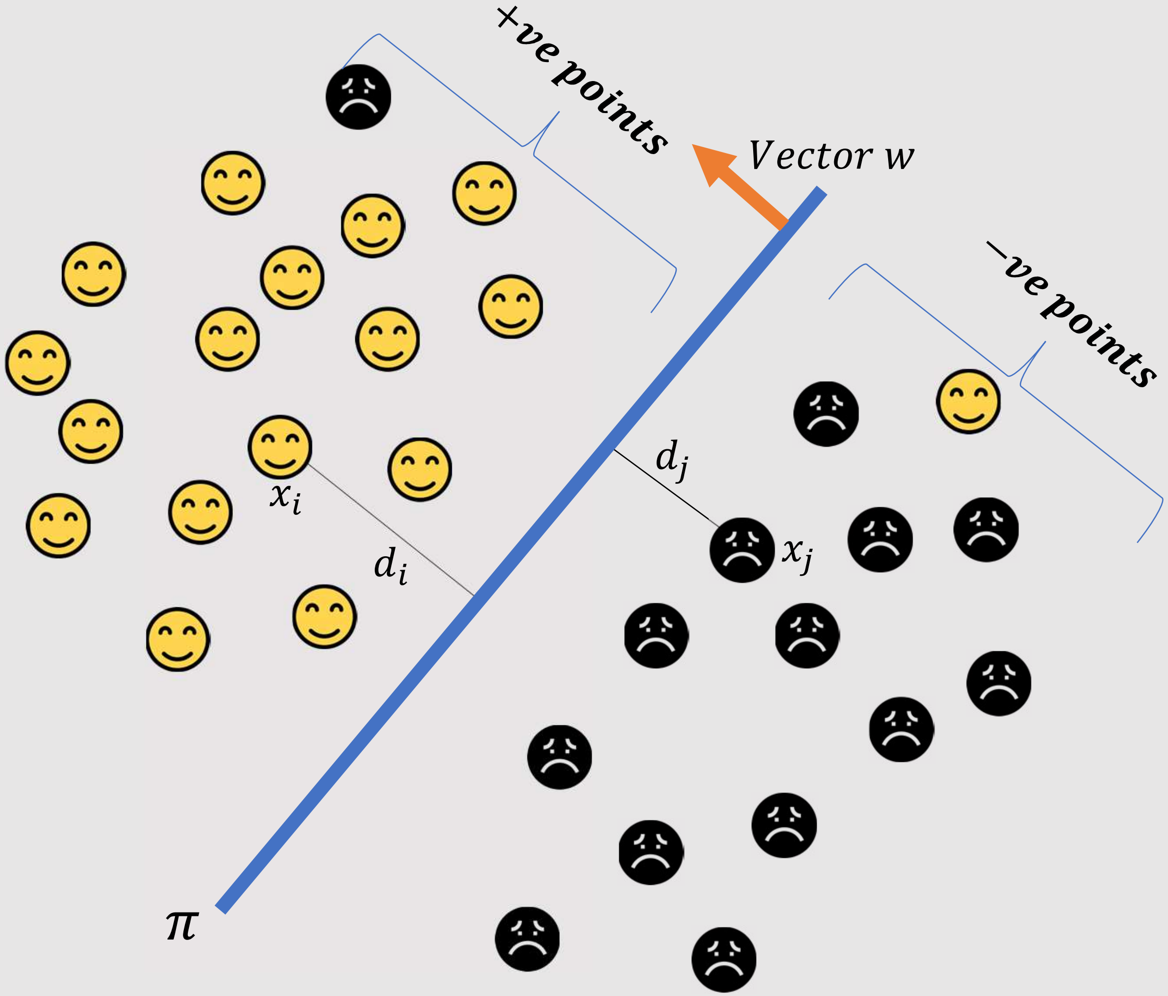
Geometric Intuition



Key Points to remember

- Biggest assumption: Data should be linearly separable or almost linearly separable
- Given the data D_n $which\ has + ve\ and\ - ve\ points$
- The task is to find the Equation of the plane that is
  - $\pi = w^T x + b = 0$
  - $w\ is\ normal\ to\ the\ plane\ - Vector$
  - $x\ is\ the\ data\ point\ - Vector$
  - $b\ is\ the\ intercept\ - Scalar$
  - It should best separates the +ve/-ve points
  - If the equation is passing from the origin then $b = 0$

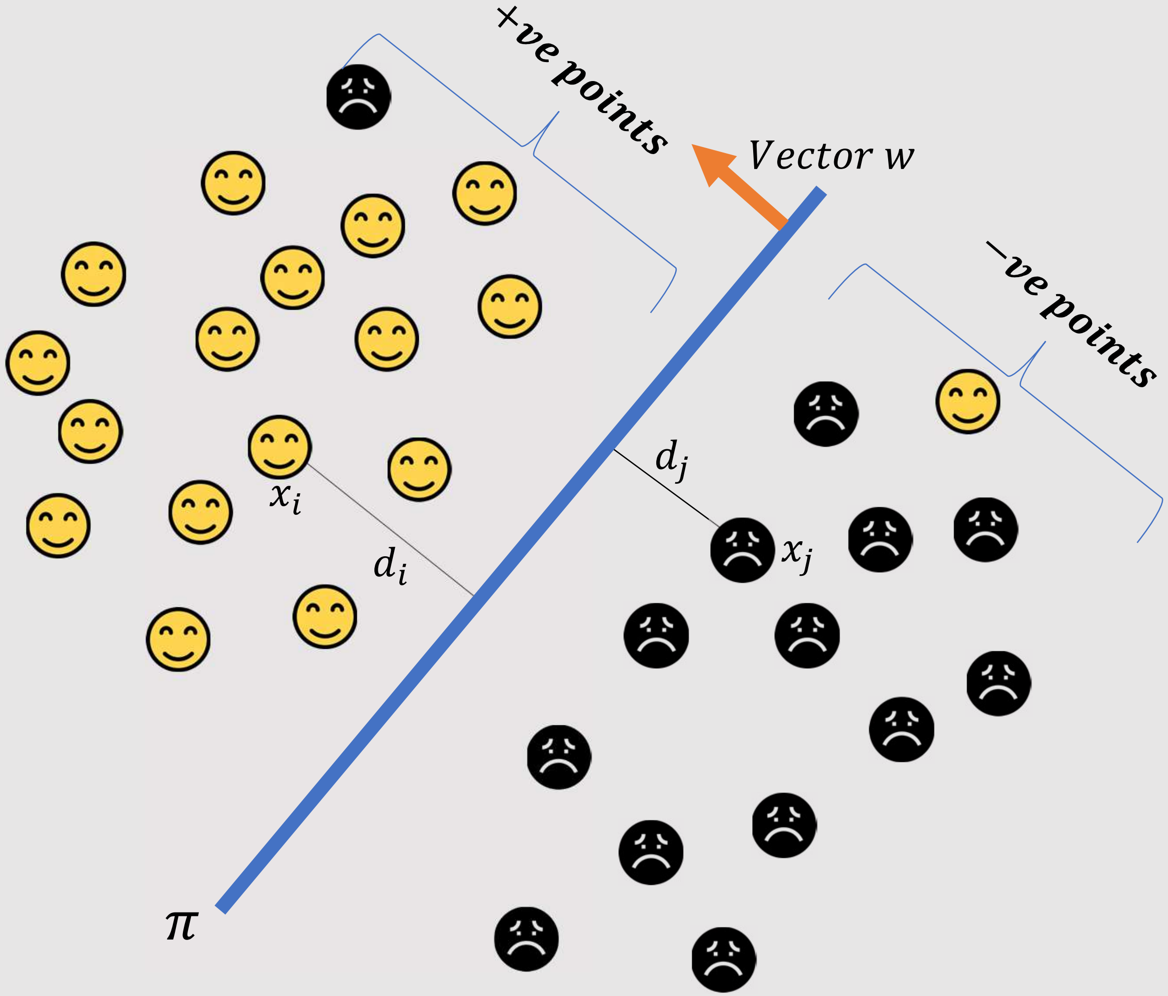
Geometric Intuition



Key Points to remember

- $y_i = +1\ for\ Positive\ points$
- $y_i = -1\ for\ Negative\ points$
- $d_i = \frac{w^T x_i}{||w||}$
  - Lets assume ||w|| is a unit vector i.e. the value is 1
- $d_i = w^T x_i$
- $d_i = w^T x_i > 0,\ as\ w\ and\ x_i\ are\ on\ same\ side\ of\ plane$
- $d_j = w^T x_j < 0, as\ w\ and\ x_j\ are\ on\ opposite\ side$
- Classifier looks like
  - If $w^T x_i > 0\ then\ y_i = +1$
  - If $w^T x_i < 0\ then\ y_i = -1$
  - Our decision surface in Logistic Regression is a line or a plane
- **Case 1:**
- If $y_i = +1\ (+ve)\ and\ w^T x_i > 0\ then\ y_i w^T x_i > 0$
  - Plane is correctly classifying the point
- **Case 2:**
- If $y_i = -1\ (-ve)\ and\ w^T x_i < 0\ then\ y_i w^T x_i < 0$
  - Plane is correctly classifying the point
- **Case 3:**
- If $y_i = +1\ (+ve)\ and\ w^T x_i < 0\ then\ y_i w^T x_i < 0$
  - Plane is incorrectly classifying the point
- **Case 4:**
- If $y_i = -1\ (-ve)\ and\ w^T x_i > 0\ then\ y_i w^T x_i < 0$
  - Plane is incorrectly classifying the point

Geometric Intuition



Key Points to remember

- $For\ classifier\ to\ be\ very\ good\ we\ need\ to\ have$
  - Minimum # of incorrect classification
  - Maximum # of correct classification
  - i.e. $y_i w^T x_i \ > 0$ (As high as possible)
  - So in nutshell we need a plane $\pi\ or\ w\ that\ will$ Maximize $y_i w^T x_i$
- $w^* = \ Argmax\ _w \ \sum_{i=1}^{n} y_i w^T x_i$
  - Here only w is variable rest all coming from the data
  - $w^* = Optimal\ w$
- This is the optimization problem we need to solve
- **Demo:**
  - Scenario 1: Linearly Separable
  - Scenario 2: Almost Linearly Separable

Assessment

Q1. What is the assumption of Logistic Regression?

- The distance between the points are the most important factor in the Logistic Regression.
- The data should be linearly or almost linearly separable
- Logistic Regression can work on any type of data there is no assumption as such
- The points should be linear in nature

Assessment

Q2. Which if the following condition is True?

- If $y_i = +1\ (+ve)\ and\ w^T x_i > 0\ then\ y_i w^T x_i > 0$, Plane is correctly classifying the point
- If $y_i = -1\ (-ve)\ and\ w^T x_i > 0\ then\ y_i w^T x_i > 0$, Plane is correctly classifying the point
- If $y_i = +1\ (+ve)\ and\ w^T x_i < 0\ then\ y_i w^T x_i > 0$, Plane is correctly classifying the point
- If $y_i = +1\ (+ve)\ and\ w^T x_i < 0\ then\ y_i w^T x_i < 0$, Plane is correctly classifying the point

Assessment

Q3. In the equation of the plane $\pi = w^T x + b = 0$, which option is correct about w, x and b?

- w and x are vectors and b is a scalar
- w is a scalar, x and b are vectors
- w and b are scalars and x is a vector
- w and b are vectors and x is a scalar

Logistic Regression – Geometric Intuition Cheat Sheet

Key Points to remember - 1

- Biggest assumption: Data should be linearly separable or almost linearly separable
- Given the data D_n $which\ has + ve\ and\ - ve\ points$
- The task is to find the Equation of the plane that is
  - $\pi = w^T x + b = 0$
  - $w\ is\ normal\ to\ the\ plane\ - Vector$
  - $x\ is\ the\ data\ point\ - Vector$
  - $b\ is\ the\ intercept\ - Scalar$
  - It should best separates the +ve/-ve points
  - If the equation is passing from the origin then $b = 0$

Key Points to remember - 2

- $y_i = +1\ for\ Positive\ points$
- $y_i = -1\ for\ Negative\ points$
- $d_i = \frac{w^T x_i}{||w||}$
  - Where ||w|| is a unit vector i.e. the value is 1
- $d_i = w^T x_i$
- $d_i = w^T x_i > 0,\ as\ w\ and\ x_i\ are\ on\ same\ side\ of\ plane$
- $d_j = w^T x_j < 0, as\ w\ and\ x_j\ are\ on\ opposite\ side$

- Classifier looks like
  - If $w^T x_i > 0\ then\ y_i = +1$
  - If $w^T x_i < 0\ then\ y_i = -1$
  - Our decision surface in Logistic Regression is a line or a plane

- **Case 1:**
- If $y_i = +1\ (+ve)\ and\ w^T x_i > 0\ then\ y_i w^T x_i > 0$
  - Plane is correctly classifying the point

- **Case 2:**
- If $y_i = -1\ (-ve)\ and\ w^T x_i < 0\ then\ y_i w^T x_i < 0$
  - Plane is correctly classifying the point

- **Case 3:**
- If $y_i = +1\ (+ve)\ and\ w^T x_i < 0\ then\ y_i w^T x_i < 0$
  - Plane is incorrectly classifying the point

- **Case 4:**
- If $y_i = -1\ (-ve)\ and\ w^T x_i > 0\ then\ y_i w^T x_i < 0$
  - Plane is incorrectly classifying the point

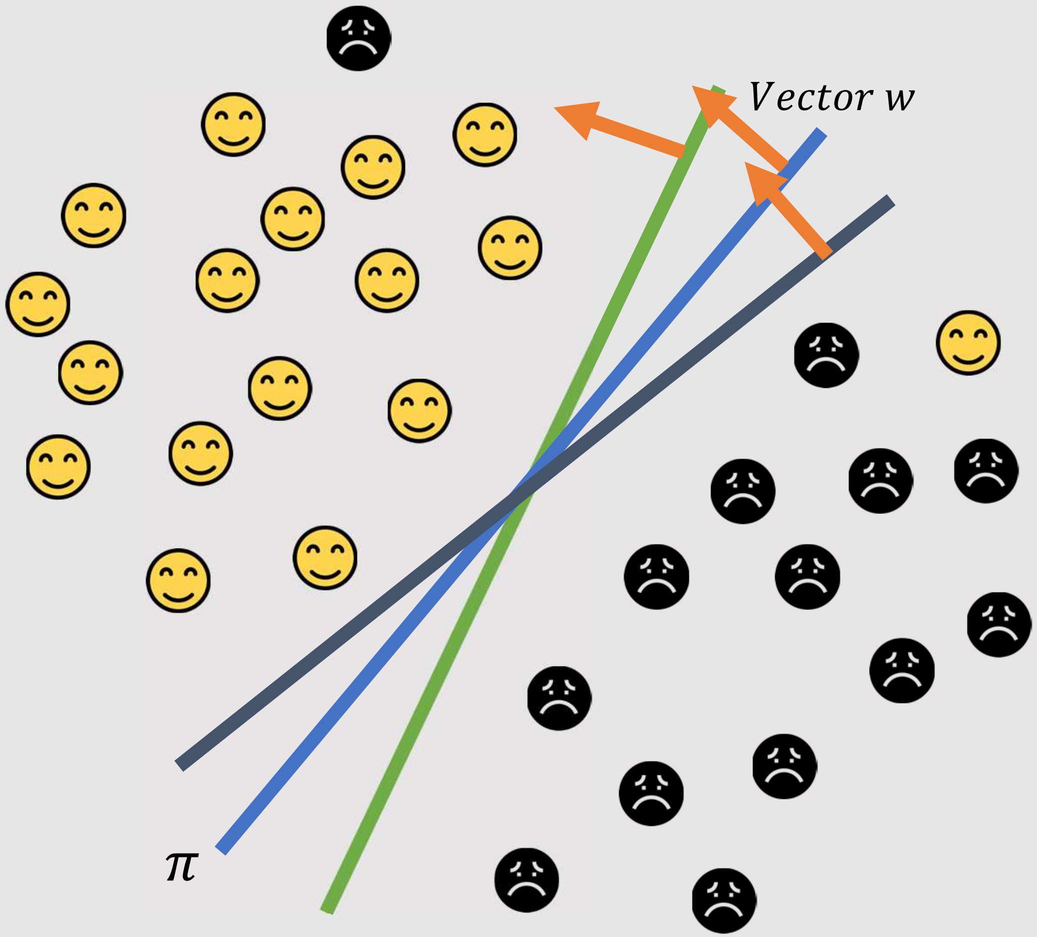
Key Points to remember - 3

- $For\ classifier\ to\ be\ very\ good\ we\ need\ to\ have$
  - Minimum # of incorrect classification
  - Maximum # of correct classification
  - i.e. $y_i w^T x_i > 0$ (As high as possible)
  - So in nutshell we need a plane $\pi\ or\ w\ that\ will$ Maximize $y_i w^T x_i$
- $w^* = Argmax_{\ w} \sum_{i=1}^{n} y_i w^T x_i$
  - Here only w is variable rest all coming from the data
  - $w^* = Optimal\ w$
- This is the optimization problem we need to solve

Sigmoid Function

Squashing

Quick Understanding

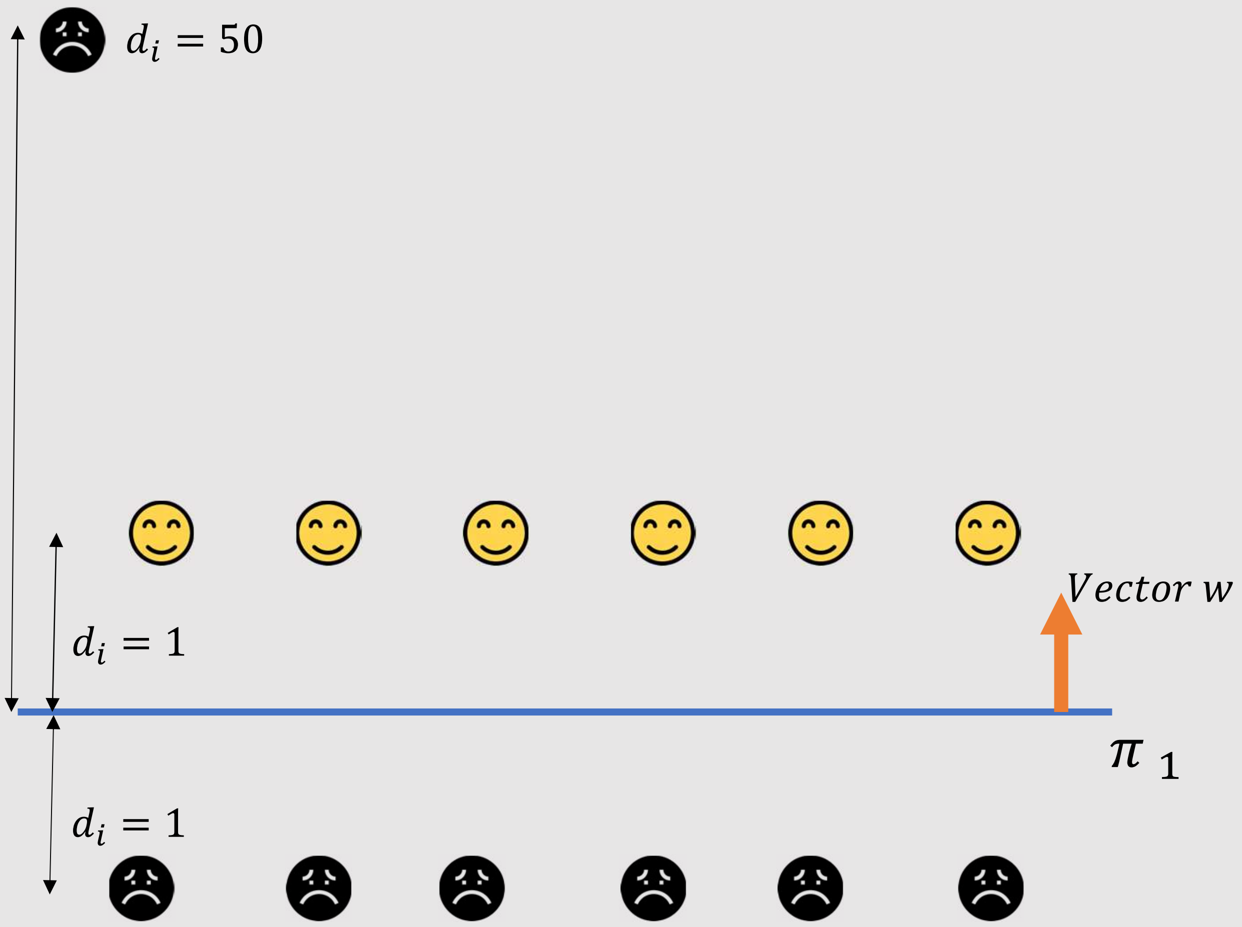


Key Points to remember

- $w^* = Argmax_{\ w} \sum_{i=1}^{n} y_i w^T x_i$
    - Here only w is variable rest all coming from the data
    - $w^* = Optimal\ w$

- $y_i w^T x_i$ +ve then correctly classified points
- $y_i w^T x_i$ - ve then incorrectly classified points
- $\boldsymbol{y_i w^T x_i}$ **(Lets call it as signed distance)**

Case 1: Fail Scenarios – Impact of Outliers

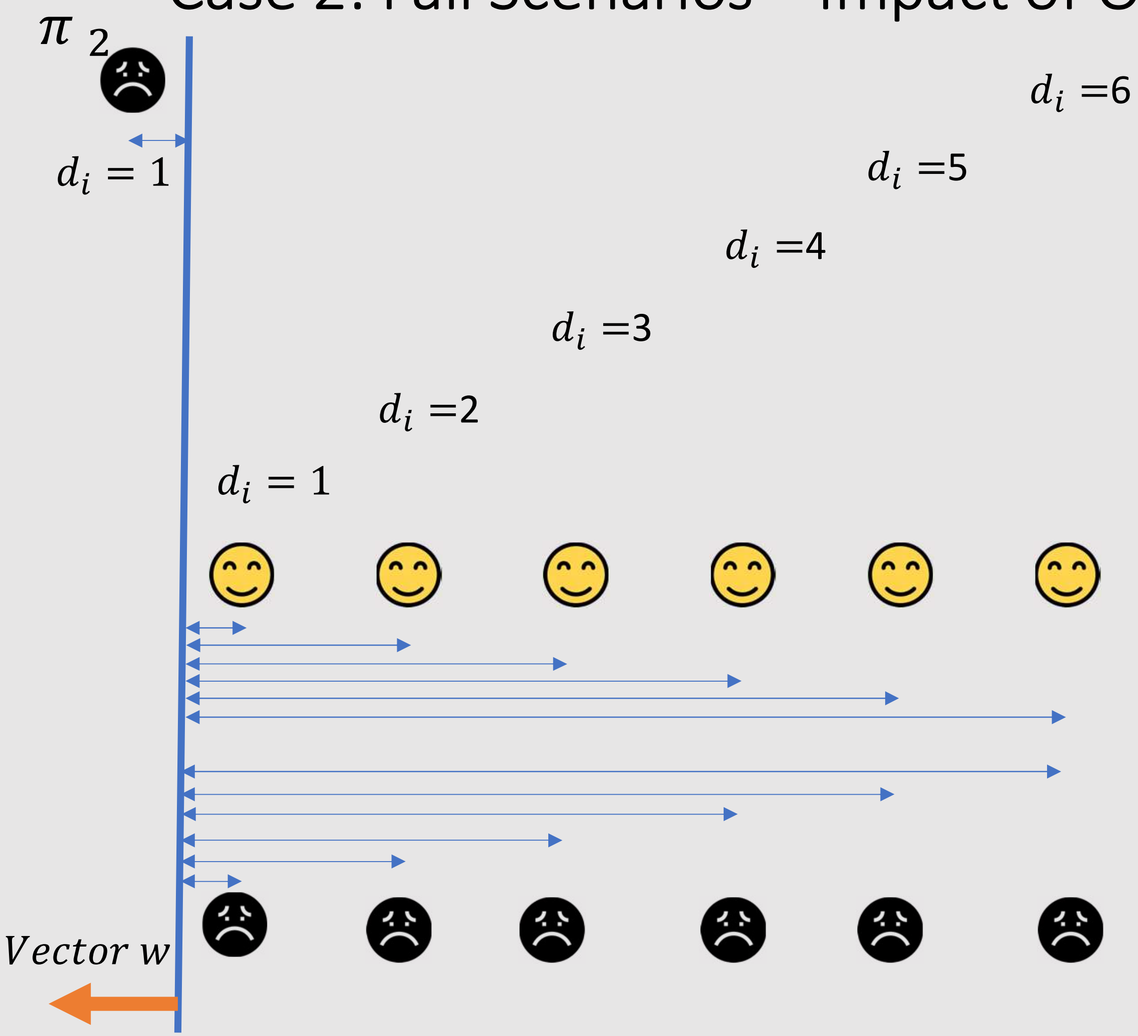


Key Points to remember

- $w^* = Argmax_{\ w} \sum_{i=1}^{n} y_i w^T x_i$
  - Here only w is variable rest all coming from the data
  - $w^* = Optimal\ w$

- $y_i w^T x_i$ +ve then correctly classified points
- $y_i w^T x_i$ - ve then incorrectly classified points
- $\boldsymbol{y_i w^T x_i}$ **(Lets call it as signed distance)**

Case 2: Fail Scenarios – Impact of Outliers



Key Points to remember

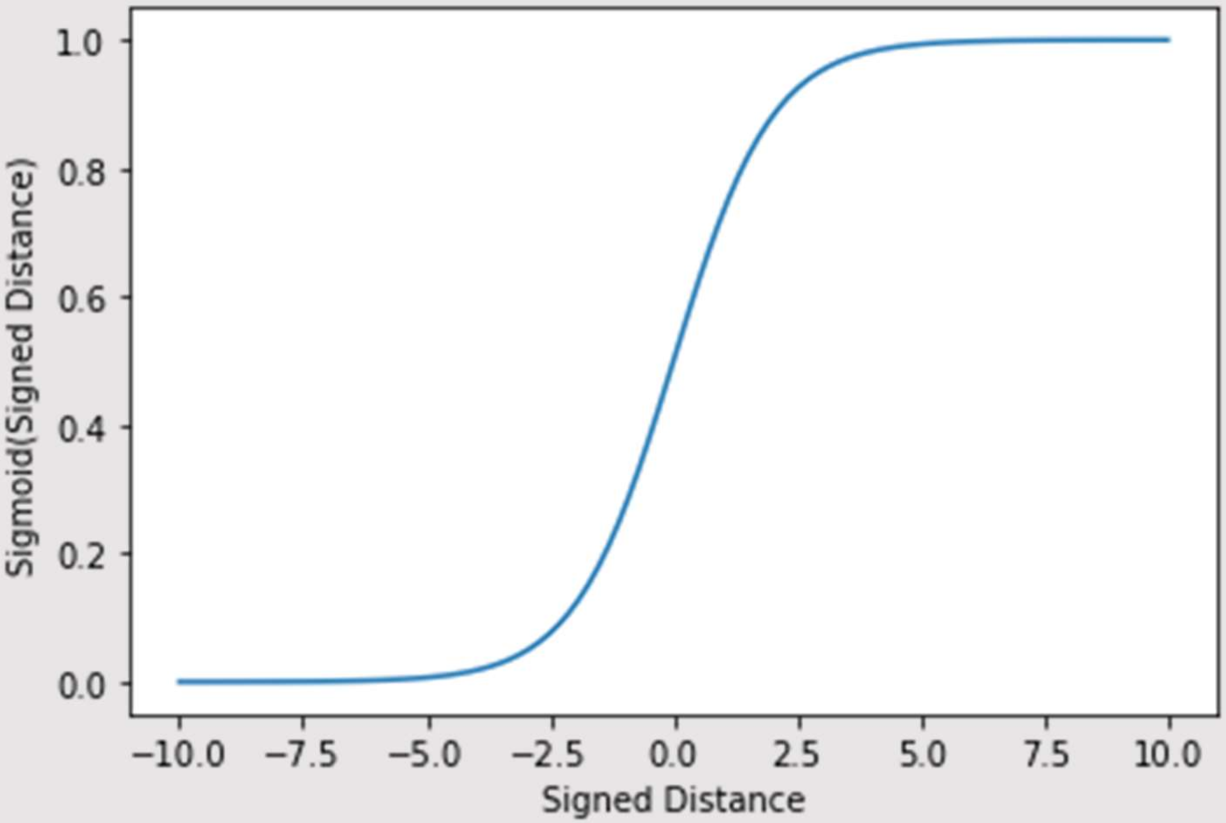
- $w^* = Argmax_w \sum_{i=1}^{n} y_i w^T x_i$
  - Here only w is variable rest all coming from the data
  - $w^* = Optimal\ w$

- $y_i w^T x_i$ +ve then correctly classified points
- $y_i w^T x_i$ - ve then incorrectly classified points
- $\boldsymbol{y_i w^T x_i}$ **(Lets call it as signed distance)**

- **Conclusion from both the scenarios is that π_2 is the best classifier which is obviously wrong.**
- **Because of the outlier this has happened.**
- **Maximizing the sum of signed distances is not outlier prone**

- **Demo:**
  - **Scenario 3**

Squashing



Key Points to remember

- Key idea of Squashing:
    - If signed distance is small use it as is, if the signed distance is large make it a smaller value
- We are converting $w^* = Argmax_{\ w} \sum_{i=1}^{n} y_i w^T x_i$ to $w^* = Argmax_{\ w} \sum_{i=1}^{n} f(y_i w^T x_i)$
- Sigmoid function is one of the functions which we use
    - $\sigma(x) = \frac{1}{1+e^{-x}}$
- $\sigma(x) = \frac{1}{1+e^{-y_i w^T x_i}}$
- $\sigma(0) = 0.5$
- It has a nice probabilistic interpretation
    - If point lies in the hyper plane the probability of $P(y_i = 1) = 0.5$
    - If point lies very far from hyper plane and towards w then the probability of $P(y_i = 1) = 0.9999$
    - If point lies very far from hyper plane and opposite of w then the probability of $P(y_i = 1) = 0$
- New optimization equation will be
    - $w^* = Argmax_{\ w} \sum_{i=1}^{n} \frac{1}{1+e^{-y_i w^T x_i}}$
- Why are we going with Sigmoid function?
    - Sigmoid function is differentiable
    - It has a nice probabilistic interpretation

Assessment

Q1. What is problem with the signed distance approach?

- It is impacted highly with the outliers
- There is no issue with signed distance it is just to make things faster we are going with Sigmoid approach
- Signed distance slows down the performance of the model
- None of the above

Assessment

Q2. Why are we using Sigmoid transformation?

- Sigmoid transformation has nice probabilistic interpretation
- Sigmoid transformation comes up with good predictions when outliers are present in the dataset
- Sigmoid function is differentiable
- All of the above

Logistic Regression – Squashing Cheat Sheet

Key Points to remember

- $w^* = Argmax_{\ w} \sum_{i=1}^{n} y_i w^T x_i$
    - Here only w is variable rest all coming from the data
    - $w^* = Optimal\ w$

- $y_i w^T x_i$ +ve then correctly classified points
- $y_i w^T x_i$ - ve then incorrectly classified points
- $\boldsymbol{y_i w^T x_i}$ **(Lets call it as signed distance)**

- **Conclusion from both the scenarios is that $\boldsymbol{\pi_2}$ is the best classifier which is obviously wrong.**
- **Because of the outlier this has happened.**
- **Maximizing the sum of signed distances is not outlier prone**

- Key idea of Squashing:
    - If signed distance is small use it as is, if the signed distance is large make it a smaller value

- We are converting $w^* = Argmax_{\ w} \sum_{i=1}^{n} y_i w^T x_i$ to $w^* = Argmax_{\ w} \sum_{i=1}^{n} f(y_i w^T x_i)$

- Sigmoid function is one of the functions which we use
    - $\sigma(x) = \frac{1}{1+e^{-x}}$

- $\sigma(x) = \frac{1}{1+e^{-y_i w^T x_i}}$

- $\sigma(0) = 0.5$

- It has a nice probabilistic interpretation
    - If point lies in the hyper plane the probability of P($y_i = 1) = 0.5$
    - If point lies very far from hyper plane and towards w then the probability of P($y_i = 1) = 0.9999$
    - If point lies very far from hyper plane and opposite of w then the probability of P($y_i = 1) = 0$

- New optimization equation will be
    - $w^* = Argmax_{\ w} \sum_{i=1}^{n} \frac{1}{1+e^{-y_i w^T x_i}}$

- Why are we going with Sigmoid function?
    - Sigmoid function is differentiable
    - It has a nice probabilistic interpretation

Mathematical Formulation

Objective Function

Optimization Problem

$$w^* = Argmax_{\ w} \sum_{i=1}^{n} \frac{1}{1+ e^{-y_i w^T x_i}}$$

$$w^* = Argmax_{\ w} \log(\sum_{i=1}^{n} \frac{1}{1+ e^{-y_i w^T x_i}})$$

$$w^* = Argmax_{\ w} - \log(\sum_{i=1}^{n} 1 + e^{-y_i w^T x_i})$$

$$w^* = Argmin_{\ w} \log(\sum_{i=1}^{n} 1 + e^{-y_i w^T x_i})$$

If 1 is not there then it is nothing but signed distance

- $y_i = +1 \ for \ Positive \ points$
- $y_i = -1 \ for \ Negative \ points$

$$w^* = Argmin_{\ w} \sum_{i=1}^{n} -y_i \log(P_i) - (1 - y_i)\log(1 - P_i)$$

$P_i = \sigma(w^T x_i)$, here $y_i = 0/1$, *above equation is probabilistic approach*

Key Points to remember

- Idea of Monotonic Functions
  - A function $g(x)$ is considered as Monotonic when $x \ increases \ then \ g(x) \ also \ increases.$
  - If $x_1 > x_2 \ then \ g(x_1) > g(x_2)$ then it monotonically increasing function
- $Log(x) \ is \ monotonically \ increasing \ function$
  - When value of $x > 0$,
  - $Log(0) \ or \ -ve \ values \ not \ defined$
  - $\boldsymbol{Log\left(\frac{1}{x}\right) = -\log(x)}$
- Simple Optimization Problem
  - $x^* = argmin_{\ x} \ x^2$
  - Here we need to find the best x which will minimize x^2
  - Minima's and Maxima's concept – Here we see 0
  - $x^* = 0$
  - $x^2 \ is \ monotonically \ increasing \ when \ x > 0$
  - $x^2 \ is \ monotonically \ decreasing \ when \ x < 0$
- Applying $Log(f(x))$
  - $x' = argmin_{\ x} \ g(f(x))$
  - $x' = argmin_{\ x} \log(x^2)$
  - Our claim is $x^* = x'$ because $g(x)$ is monotonic function
  - If $g(x)$ is a monotonic function (Inc./Dec.) then
    - $argmin_{\ x} f(x) = argmin_{\ x} g(f(x))$
    - $argmax_{\ x} f(x) = argmax_{\ x} g(f(x))$
  - $\boldsymbol{argmax_{\ x} f(x) = argmin_{\ x} - f(x)}$
  - $\boldsymbol{argmax_{\ x} - f(x) = argmin_{\ x} f(x)}$
- Use below plots for explanation
  - $Log(x)$
  - x^2
  - $Log(x^2)$
  - $\boldsymbol{-x^2}$

Assessment

Q1. What is the property of monotonic function?

- If $\mathrm{f}(x) increases\ then\ g\big(f(x)\big)\ decreases$
- If $\mathrm{f}(x) increases\ then\ g\big(f(x)\big)\ will\ be\ constant$
- If $\mathrm{f}(x) increases\ then\ g\big(f(x)\big)\ increases$
- If $\mathrm{f}(x) increases\ then\ g\big(f(x)\big)\ becomes\ 0$

Assessment

Q2. In the geometric interpretation on Logistic regression what is the value of y_i?

- $y_i\ varies\ from\ 0\ to\ 1$
- $y_i\ varies\ from\ -1\ to\ 1$
- $y_i\ varies\ from\ -\infty\ to\ 1$
- $y_i\ varies\ from\ -\infty\ to+\infty$

Logistic Regression - Optimization Problem Cheat Sheet

Key Points to remember

- Idea of Monotonic Functions
  - A function $g(x)$ is considered as Monotonic when x *increases then* $g(x)$ *also increases.*
  - If $x_1 > x_2$ *then* $g(x_1) > g(x_2)$ then it monotonically increasing function

- $Log(x)$ *is monotonically increasing function*
  - When value of $x > 0$,
  - $Log(0)$ *or* $-$ *ve values not defined*
  - $\boldsymbol{Log\left(\frac{1}{x}\right) = -\log(x)}$

- Simple Optimization Problem
  - $x^* = argmin\ _x\ x^2$
  - Here we need to find the best x which will minimize x^2
  - Minima's and Maxima's concept – Here we see 0
  - $x^* = 0$
  - x^2 *is monotonically increasing when* $x > 0$
  - x^2 *is monotonically decreasing when* $x < 0$

- Applying $Log(f(x))$
  - $x' = argmin\ _x\ g(f(x\ ))$
  - $x' = argmin\ _x \log(x^2)$
  - Our claim is $x^* = x'$ because $g(x)$ is monotonic function
  - If $g(x)$ is a monotonic function (Inc./Dec.) then
    - $argmin\ _x f(x) = argmin\ _x g(f(x))$
    - $argmax\ _x f(x) = argmax\ _x g(f(x))$
  - $\boldsymbol{argmax\ _x f(x) = argmin\ _x - f(x)}$
  - $\boldsymbol{argmax\ _x - f(x) = argmin\ _x f(x)}$
- Use below plots for explanation
  - $Log(x)$
  - x^2
  - $Log(x^2)$
  - $\boldsymbol{-x^2}$

Weight Vector

How to interpret?

Weight Vector

$$w^* = Argmin_{\ w} \log(\sum_{i=1}^{n} 1 + e^{-y_i w^T x_i})$$

Weight Vector

Size of weight vector same as number of dimensions

Key Points to remember

- Every feature will have a weight associated with it
- Geometric Interpretation:
  - Classifier looks like
    - If $w^T x_i > 0 \; then \; y_i = +1$
    - If $w^T x_i < 0 \; then \; y_i = -1$
- Probabilistic Interpretation:
  - $\sigma(w^T x_i) = P(y_q = 1)$

- Interpretation of w:
  - Case 1:
    - If $w_i = \; +ve, x_{qi} is \; increasing$
      - Then $w_i x_{qi} \; increases$
      - That means $\sigma\left(w^T x_{qi}\right) \; increases$
      - $P\left(y_q = 1\right) \; increases$
  - Case 2:
    - If $w_i = -ve, x_{qi} is \; increasing$
      - Then $w_i x_{qi} \; decreases$
      - That means $\sigma\left(w^T x_{qi}\right) \; decreases$
      - $P\left(y_q = 1\right) \; decreases$
      - $P\left(y_q = -1\right) \; increases$

Assessment

Q1. When the weight $w_i\ is + ve\ and\ x_{qi}$ is low what will be the probability of $P(y_i = 1)$?

- High
- Low
- Towards 0.5

Assessment

Q2. When the weight $w_i\ is + ve\ and\ x_{qi}$ is high what will be the probability of $P(y_i = 1)$?

- High
- Low
- Towards 0.5

Assessment

Q3. When the weight $w_i\ is\ -ve\ and\ x_{qi}$ is high what will be the probability of $P(y_i = 1)$?

- High
- Low
- Towards 0.5

Logistic Regression – Weight Vector Cheat Sheet

Key Points to remember

- Every feature will have a weight associated with it
- Geometric Interpretation:
    - Classifier looks like
        - If $w^T x_i > 0 \; then \; y_i = +1$
        - If $w^T x_i < 0 \; then \; y_i = -1$
- Probabilistic Interpretation:
    - $\sigma(w^T x_i) = P(y_q = 1)$

- Interpretation of w:
    - Case 1:
        - If $w_i = \; +ve, x_{qi} is \; increasing$
            - Then $w_i x_{qi} \; increases$
            - That means $\sigma\left(w^T x_{qi}\right) increases$
            - $P\left(y_q = 1\right) increases$

    - Case 2:
        - If $w_i = -ve, x_{qi} is \; increasing$
            - Then $w_i x_{qi} \; decreases$
            - That means $\sigma\left(w^T x_{qi}\right) decreases$
            - $P\left(y_q = 1\right) decreases$
            - $P\left(y_q = -1\right) increases$

L2 Regularization

Overfitting vs. Underfitting

L2 Regularization (Ridge Regression)

$$w^* = Argmin_{\ w} \log(\sum_{i=1}^{n} 1 + e^{-y_i w^T x_i})$$

Weight Vector

Let's say $y_i w^T x_i = z_i$

$$w^* = Argmin_{\ w} \log(\sum_{i=1}^{n} 1 + e^{-z_i}) \geq 0$$

$$w^* = Argmin_{\ w} \log(\sum_{i=1}^{n} 1 + e^{-z_i}) + \lambda ||w||_2^{\ 2}$$

Tug of war

Key Points to remember

- Plot e^{-z}
  - $e^{-z} \geq 0$
- $Log(1 + e^{-z}) \geq 0$
  - $Log(1) = 0$
  - $Log(1 + \delta) \geq Log(1)$
    - $\delta \geq 0$
- $Minimum\ value\ of\ optimization\ equation\ will\ be\ 0$
- If $z_i = +ve$ and $Z_i \rightarrow \infty$ (for all i)
  - Then $e^{-z_i} \rightarrow 0$
  - Then $Log(1 + e^{-z_i}) \rightarrow 0$
- $y_i w^T x_i = z_i$ -> Here the only variable is w
  - We need to modify w a way that each $z_i \rightarrow +\infty$
    - z_i = +ve; x_i is correctly classified by w
    - z_i -> +∞
    - For $z_i \rightarrow +\infty, we\ have\ to\ have\ w_i \rightarrow +\infty/-\infty$
    - $w_i\ is\ becoming\ very\ large$
- Alert: What is we have outliers?
  - This will lead to overfitting
- We have to use one key aspect that is w is normal
  - i.e. $w^T w = 1$
- **L2 Regularization**
  - $w^* = Argmin_{\ w} \log(\sum_{i=1}^{n} 1 + e^{-z_i}) + \lambda w^T w$
  - $w^* = Argmin_{\ w} \log(\sum_{i=1}^{n} 1 + e^{-z_i}) + \lambda \sum_{j=1}^{d} w_j^{\ 2}$
  - $w^* = Argmin_{\ w} \log(\sum_{i=1}^{n} 1 + e^{-z_i}) + \lambda ||w||_2^{\ 2}$
  - Square of L2 norm of w
  - 2nd part of equation (Regularization term) will ensure w will not reach to $+\infty/-\infty$
  - λ is a hyper parameter
    - $\lambda = 0, Overfit$
    - $\lambda = Very\ Large, Underfit$

Assessment

Q1. Why regularization is required?

- So that the model will not be underfit
- So that the model will not be overfit
- So that the model will not be overfit as well as underfit
- None of the above

Assessment

Q2. What will happen when the value of λ is equal to 0?

- The model will be underfit
- The model will be overfit
- λ is not really taking care of overfitting and underfitting
- None of the above

Assessment

Q3. What will happen when the value of λ is very high?

- The model will be underfit
- The model will be overfit
- λ is not really taking care of overfitting and underfitting
- None of the above

Logistic Regression – L2 Regularization Cheat Sheet

Key Points to remember

- Plot e^{-z}
  - $e^{-z} \geq 0$
- $Log(1 + e^{-z}) \geq 0$
  - $Log(1) = 0$
  - $Log(1 + \delta) \geq Log(1)$
    - $\delta \geq 0$
- $Minimum\ value\ of\ optimization\ equation\ will\ be\ 0$
- If $z_i = +ve$ and $Z_i \rightarrow \infty$ (for all i)
  - Then $e^{-z_i} \rightarrow 0$
  - Then $Log(1 + e^{-z_i}) \rightarrow 0$
- $y_i w^T x_i = z_i$ -> Here the only variable is w
  - We need to modify w a way that each $z_i \rightarrow +\infty$
    - z_i = +ve; x_i is correctly classified by w
    - z_i -> +∞
    - For $z_i \rightarrow +\infty, we\ have\ to\ have\ w_i \rightarrow +\infty/-\infty$
    - $w_i\ is\ becoming\ very\ large$
- Alert: What is we have outliers?
  - This will lead to overfitting
- We have to use one key aspect that is w is normal
  - i.e. $w^T w = 1$

- **L2 Regularization**
  - $w^* = Argmin\ _w\ \log(\sum_{i=1}^{n} 1 + e^{-z_i})$ + $\lambda w^T w$
  - $w^* = Argmin\ _w\ \log(\sum_{i=1}^{n} 1 + e^{-z_i})$ + $\lambda \sum_{j=1}^{d} w_j{}^2$
  - $w^* = Argmin\ _w\ \log(\sum_{i=1}^{n} 1 + e^{-z_i})$ + $\lambda ||w||_2{}^2$
  - Square of L2 norm of w
  - 2nd part of equation (Regularization term) will ensure w will not reach to $+\infty/-\infty$
  - λ is a hyper parameter
    - $\lambda = 0, Overfit$
    - $\lambda = Very\ Large, Underfit$

L1 Regularization

Understand the sparsity

L1 Regularization (Lasso Regression)

$$w^* = Argmin_{\ w} \log(\sum_{i=1}^{n} 1 + e^{-z_i}) + \lambda||w||_1$$

Key Points to remember

- **Are there any alternatives of L2 Regularization?**

- **L1 Regularization**
    - $w^* = Argmin_{\ w} \log(\sum_{i=1}^{n} 1 + e^{-z_i}) + \lambda|w|$
    - $w^* = Argmin_{\ w} \log(\sum_{i=1}^{n} 1 + e^{-z_i}) + \lambda \sum_{j=1}^{d} |w_j|$
    - $w^* = Argmin_{\ w} \log(\sum_{i=1}^{n} 1 + e^{-z_i}) + \lambda||w||_1$
    - L1 norm of w
    - 2nd part of equation (Regularization term) will ensure w will not reach to $+\infty/-\infty$
    - λ is a hyper parameter
        - $\lambda = 0, Overfit$
        - $\lambda = Very\ Large, Underfit$

- If there is a vector w = <$w_1, w_2, w_3 \ldots\ldots w_d$>
- Solution of Logistic Regression is sparse if many of $\boldsymbol{w_i's}$ are 0
- Means many unimportant features will have weights will be 0
- When we will use L2 Regularization the $\boldsymbol{w_i's}$ will be less but not 0

- Elastic-Net
- $w^* = Argmin_{\ w} \log(\sum_{i=1}^{n} 1 + e^{-z_i}) + \lambda_1||w||_1 + \lambda_2||w||_2^{\ 2}$
- Here $\lambda_1\ and\ \lambda_2$ are hyper parameters

Assessment

Q1. What is the difference between L1 Regularization and L2 Regularization?

- L1 Regularization leads to all the unimportant weights as ∞
- L2 Regularization leads to all the unimportant weights as ∞
- L2 Regularization leads to all the unimportant weights as 0
- L1 Regularization leads to all the unimportant weights as 0

Logistic Regression – L2 Regularization Cheat Sheet

Key Points to remember

- **Are there any alternatives of L2 Regularization?**

- **L1 Regularization**
    - $w^* = Argmin_{\ w} \ \log(\sum_{i=1}^{n} 1 + e^{-z_i})$ + $\lambda|w|$
    - $w^* = Argmin_{\ w} \ \log(\sum_{i=1}^{n} 1 + e^{-z_i})$ + $\lambda \sum_{j=1}^{d} |w_j|$
    - $w^* = Argmin_{\ w} \ \log(\sum_{i=1}^{n} 1 + e^{-z_i})$ + $\lambda||w||_1$
    - L1 norm of w
    - 2nd part of equation (Regularization term) will ensure w will not reach to $+\infty/-\infty$
    - λ is a hyper parameter
        - $\lambda = 0, Overfit$
        - $\lambda = Very\ Large, Underfit$

- If there is a vector w = <$w_1, w_2, w_3 \ \ldots\ldots w_d$>
- Solution of Logistic Regression is sparse if many of $\boldsymbol{w_i's}$ are 0
- Means many unimportant features will have weights will be 0
- When we will use L2 Regularization the $\boldsymbol{w_i's}$ will be less but not 0

Probabilistic Interpretation

Logistic Regression

Probabilistic Interpretation

$$w^* = Argmin_{\ w} \sum_{i=1}^{n} -y_i \log(P_i) - (1-y_i)\log(1-P_i)$$

$P_i = \sigma(w^T x_i)$, here $y_i = 0/1$, *above equation is probabilistic approach*

$$w^* = Argmin_{\ w} \log(\sum_{i=1}^{n} 1 + e^{-y_i w^T x_i})$$

here $y_i = -1/1$, *above equation is Geometric approach*

Key Points to remember

- Features are real valued and have gaussian distribution
  - i.e. $P(x_i|Y = y_k)$ is gaussian distributed with some mean and standard deviation
- Y is a random Boolean variable following Bernoulli distribution
- X_i and X_j are conditionally independent given Y
- **Case 1:**
  - **Geometric Interpretation**
    - $Y_i = +ve$
    - $\log(1 + e^{-w^T x_i})$
  - **Probabilistic Interpretation**
    - $Y_i = +ve$
    - $\log(1 + e^{-w^T x_i})$
- **Case 2:**
  - **Geometric Interpretation**
    - $Y_i = -ve$
    - $\log(1 + e^{w^T x_i})$
  - **Probabilistic Interpretation**
    - $Y_i = -ve$
    - $\log(1 + e^{w^T x_i})$

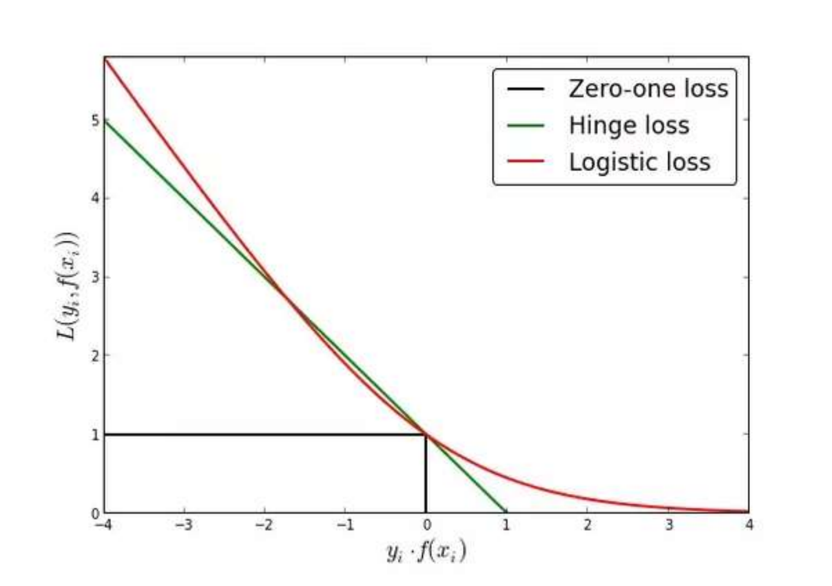
Loss Minimization Interpretation

Logistic Regression

Loss Minimization Interpretation

$$w^* = Argmin\ _w\ \log(\sum_{i=1}^{n} 1 + e^{-y_i w^T x_i}\ )$$

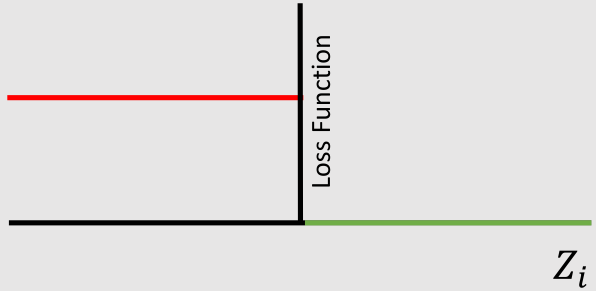
$$w^* = Argmin\ _w\ Number\ of\ incorrectly\ classified\ points$$



Source: https://www.quora.com/Why-does-the-logistic-regression-cost-function-work

Key Points to remember

- +1 – If the point is incorrectly classified
- 0 – If the point is correctly classified
- We have to minimize the loss and maximize the profit.
- 0-1 loss function
- 0-1 Loss function (Z_i) = 1 if $Z_i < 0,$ 0 if $Z_i > 0$



- The above function is not differentiable
    - The function has to be continuous
    - There is a discontinuity at $Z_i = 0$

- Let's approximate it using Logistic Loss
    - Plot in google log(1 + e^x)
    - Logistic is one of the approximations of 0-1 Loss
        - **Positive Side**
        - In 0-1 loss when $Z_i > 0$ then $0 - 1$ is 0
        - Logistic loss tends towards 0
        - **Negative Side**
        - In 0-1 loss when $Z_i < 0$ then $0 - 1$ is 1
        - Logistic loss increasing

Hyper Parameters Search

Grid Search and Random Search

Hyper Parameter Search

$$w^* = Argmin_{\ w} \log(\sum_{i=1}^{n} 1 + e^{-z_i}) + \lambda ||w||_2^{\ 2}$$

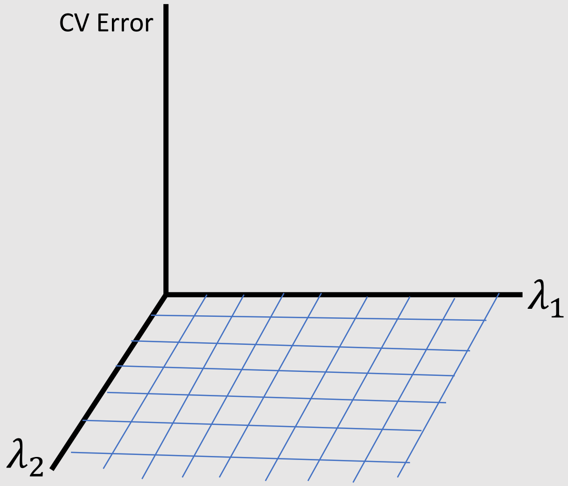
$$w^* = Argmin_{\ w} \log(\sum_{i=1}^{n} 1 + e^{-z_i}) + \lambda_1 ||w||_1 + \lambda_2 ||w||_2^{\ 2}$$

Key Points to remember

- λ = 0 -> Overfitting
- λ = High -> Underfitting

- How to determine the value of λ?
- The value of λ is a real number and the number of possible values are infinity. So to find the right λ value is Grid Search.
- We will plot all the values of λ in X axis and CV error in Y Axis
- λ = [0.001, 0.01, 0.1, 1, 10, 100, 1000] and soon
- The minimum error we will get is the best value of λ

- Let's say we have Elastic Net and here we have two Hyper Parameters



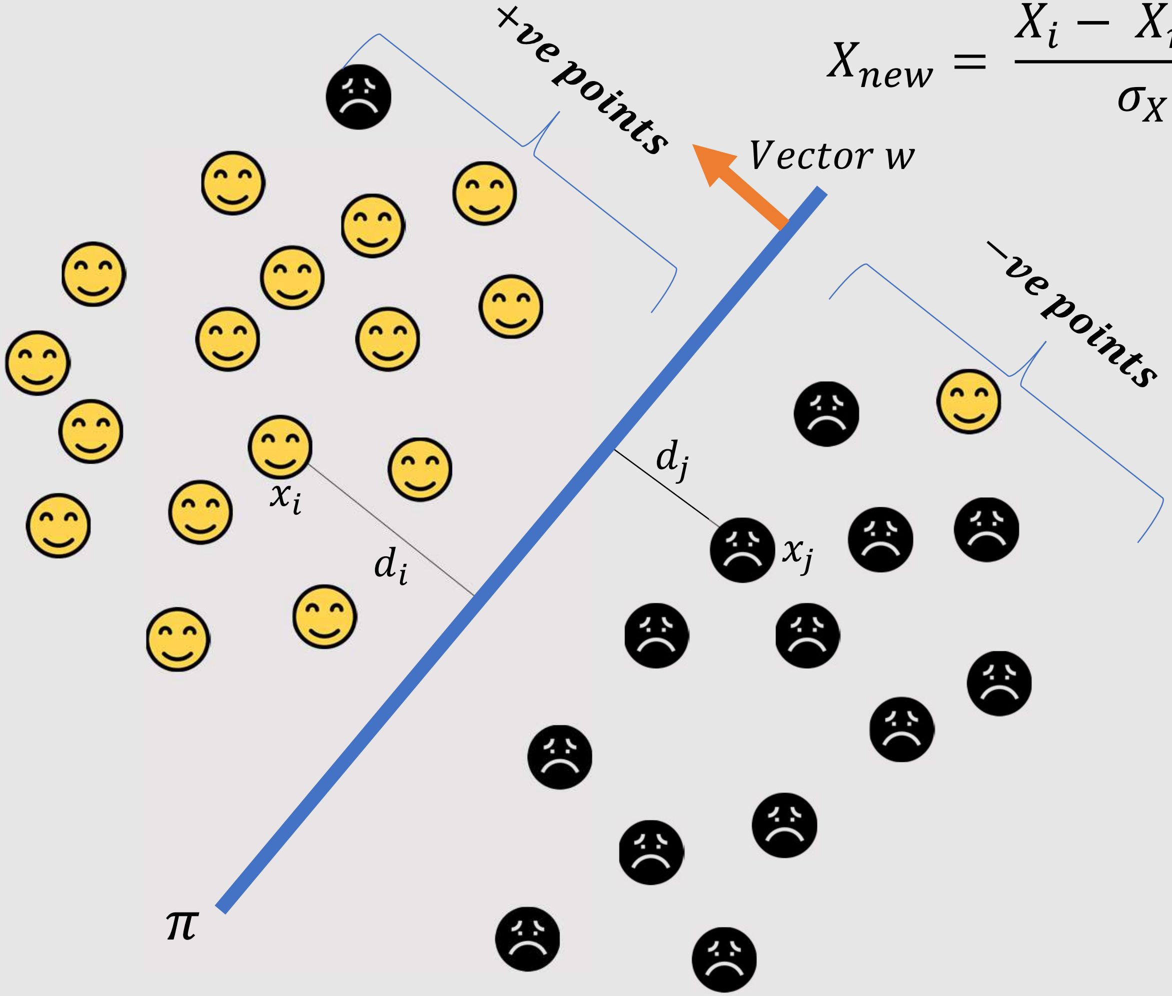
- Problem with Grid Search – More number of times need to run the algorithm. As the number of Hyper parameter increases the number of time the model needs to be train increases exponentially.
- Random Search: We can randomly pick the values from the given interval.

Column Standardization

Mandatory in Logistic Regression

Column Standardization

$$X_{new} = \frac{X_i - X_{mean}}{\sigma_X}$$



Key Points to remember

- Before training your data we need to perform the feature standardization
- Column standardization is also known as Mean centering and scaling

Feature Importance and Model Interpretability

Logistic Regression

Feature Importance

$$f \rightarrow < f_1, f_2, f_3, \dots \dots \dots \dots . f_j \dots \dots . . f_d >$$

$$w \rightarrow < w_1, w_2, w_3, \dots \dots \dots \dots . w_j \dots \dots . . w_d >$$

Key Points to remember

- Assuming that all the features are independent (Naïve Bayes) (As discussed, in the Probabilistic approach)
- We can use weights to get the feature importance
- **Case1 :**
    - |Weights| = If the absolute value is large then its contribution to $w^T x_j$ is large.
    - That means if w_j $increases$ then $w^T x_j$ increases as well.
    - The probability of a query point as positive class is higher
- **Case1 :**
    - |Weights| = If the absolute value is large then its contribution to $w^T x_j$ is large.
    - That means if w_j $-ve\ and\ Large$ then $w^T x_j$ increases as well.
    - The probability of a query point as negative class is higher
- We can determine the importance of the features in logistic regression
- Example:
    - Predict Gender: Male and Female (+1/-1)
    - Hair Length: $|w_{HL}|$ is large
    - The value of w_{HL} will be negative
    - As w_{HL} increases the probability of Female increase.

    - Height: $|w_{HT}|$ is large
    - The value of w_{HT} will be positive
    - As w_{HT} increases the probability of Male increase.
    - But this feature is not very significant then w_{HT} will be medium
- **Model Interpretability:** We can get the top features and give interpretation

Collinearity or Multicollinearity

Logistic Regression

Collinearity or Multicollinearity

Key Points to remember

- If there is a collinearity or multicollinearity weigh vectors are not useful for feature importance.
- Feature $f_1 = \alpha f_2 + \beta$
- This is nothing but Feature f_1 and f_2 are collinear
- Same way if we have more features then it is a case of multicollinearity
- Why does weight vectors are not useful for feature importance?
- Example:
- D = <x_i, y_i>$_{i=1}^{n}$
- $w^* = < 1,2,3 >$
- $w^T x_q = x_{q1} + 2x_{q2} + 3x_{q3}$
    - If $f_2 = 1.5f_1$ -> The features are collinear.
- $w^T x_q = 4x_{q1} + 3x_{q3}$
- $w^{\sim} = < 4,0,3 >$
- Both weight vectors w^* and $w^{\sim}$ will give the same classifier.
    - Our feature importance will change in both the classifier.
    - Conclusion changed drastically
    - If features are collinear/Multicollinear the weigh vector can change arbitral
- Before we use weight vector we need to find if the features are multicollinear or not
- We can go for Perturbation technique
    - Add small error (small noise)
    - Before adding a noise compute Weight vector
    - After perturbation find weight vector again
    - If these values differ drastically then we can conclude that the features are multicollinear
- We can use forward feature selection technique in the case of multicollinearity

Train and Run time Space and Time complexity

Logistic Regression

Train and Runtime Space and Time

Key Points to remember

- Training time of Logistic Regression is O(ND)
    - N = No of points in D_{Train}
    - D = Dimensionality
- Run time of Logistic Regression is O(ND)
    - We have to store only the vector W
    - We have to do $w^T x_q$ and add (Multiplications and Additions)
    - Size of vector W is D, so space complexity is O(D)
    - Time complexity is O(D) also
    - If D is small Logistic Regression is Awesome
    - For Low Latency applications (Given a point x_q) The time it should take is very low
    - Memory efficient
    - **Favorite algorithm at internet companies**
    - **What is D is large?**
        - Need more multiplications and additions
        - We can go for L1 Regularization
            - This will lead to less important features to 0
        - As $\lambda\ increases$ sparsity increases but at the same time Bias will also increases
        - We need to come up with a tradeoff between Bias, Variance and Latency.

Real World Cases

Logistic Regression

Real World Cases

Key Points to remember

- Decision surface is a Linear / Hyperplane
- Assumption we have is in Logistic Regression is Data is Linearly separable or almost linearly separable

- Feature Importance and Interpretability
  - We use $|w_j|$ if the features are not multicollinear
  - If the features are multicollinear then we can go for forward feature selection

- Imbalance Dataset
  - Up sampling and Down sampling

- Outliers
  - Less Impact because of the sigmoid function
  - But it is not completely avoided so we can do below steps
    - Take $D_{Train} \rightarrow\ w^*$ (Calculate weights)
    - We can then take x_i $and\ calculate\ w^T x_i$
      - This is the distance from π to x_i
      - Remove the points which are very far away from π the new data will be $D_{Train}{}'$
      - We will create the model again on $D_{Train}{}'$ and that will be the final solution

  - We can also go for Outlier removal or treatment

- Missing value – We can go for standard imputation

- Multiclass Extension – We normally do OVR (One Vs. Rest)

- Similarity Matrix – We can use Kernel Logistic Regression when the source is similarity matrix.

Real World Cases

Key Points to remember

- Best and Worst Cases
    - Almost or Linearly separable
    - Low Latency Requirements
    - Very Fast to train
    - If the data is not Linearly separable the worst

- High Dimensionality
    - If the D is large then the chance of Data to be linearly separable is high
    - If we want low latency system we can go for L1 regularization

Imbalance Data – Geometric View

Logistic Regression

Imbalance Dataset

Key Points to remember

- Best and Worst Cases
  - Almost or Linearly separable
  - Low Latency Requirements
  - Very Fast to train
  - If the data is not Linearly separable the worst

- High Dimensionality
  - If the D is large then the chance of Data to be linearly separable is high
  - If we want low latency system we can go for L1 regularization