
Univ.AI

Classification

Logistic Regression

Univ.AI

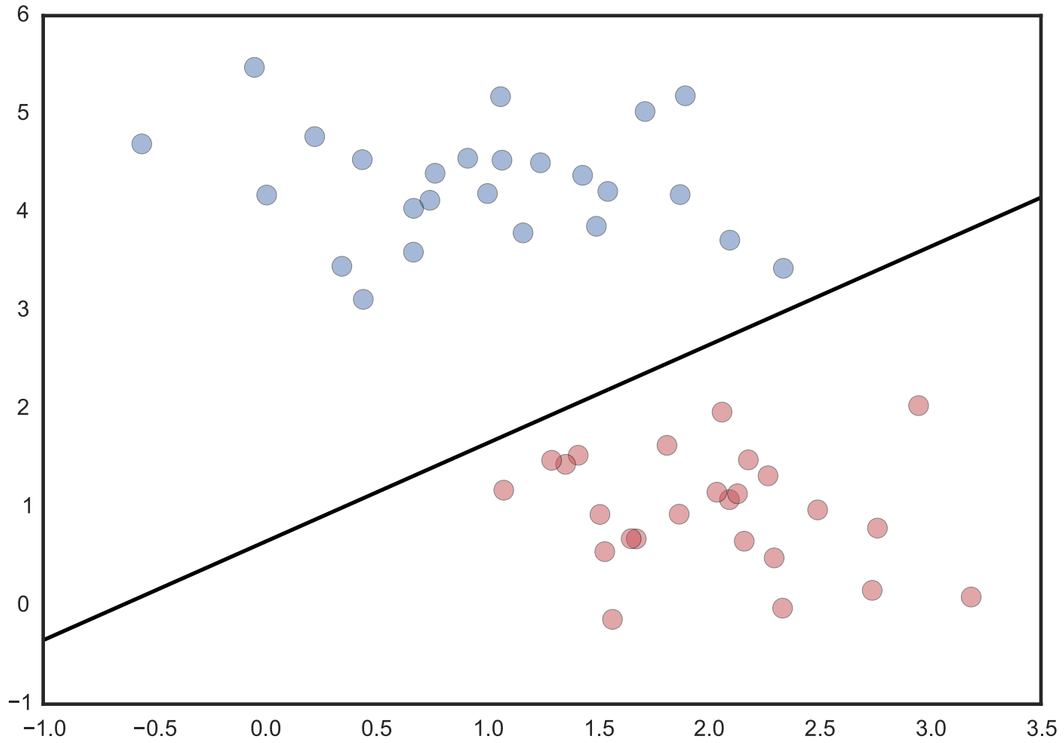
Last time

- Validation
- Cross Validation
- Regularization

This time

1. Classification
2. Logistic Regression
3. Metrics

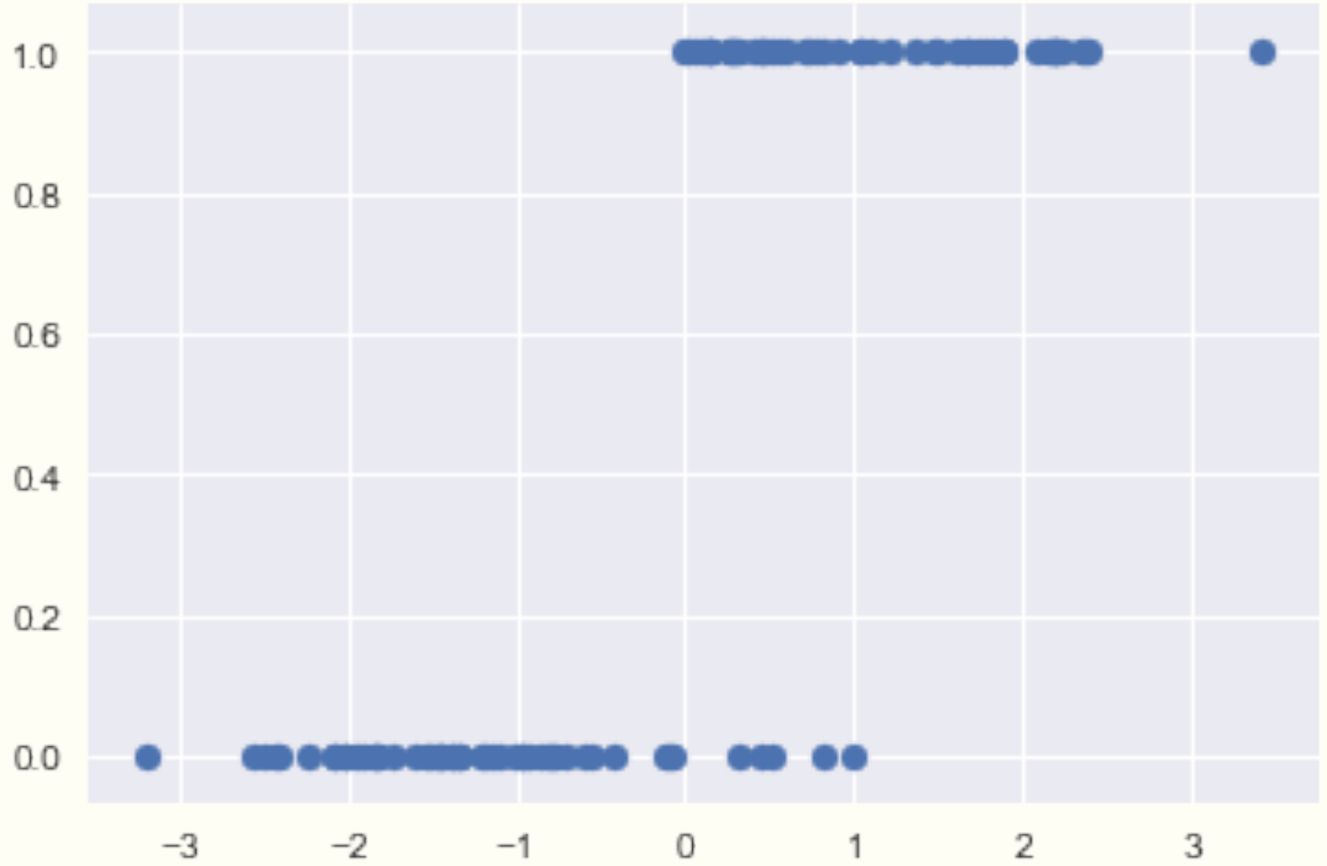
1. CLASSIFICATION



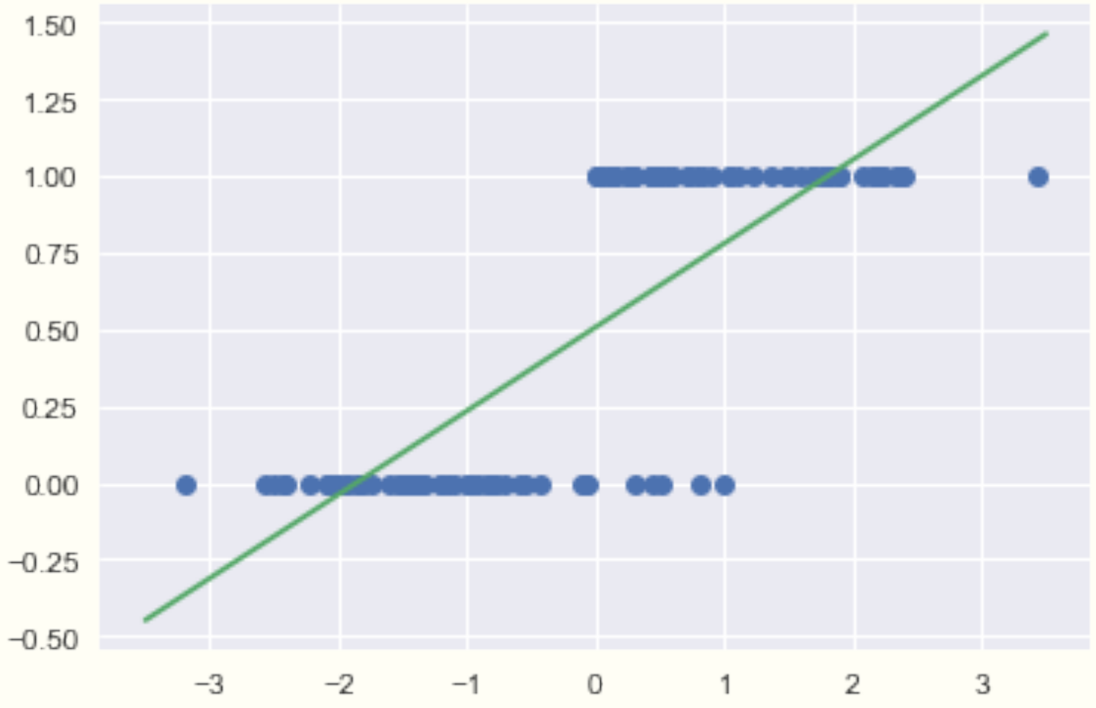
- will a customer churn?
- is this a check? For how much?
- a man or a woman?
- will this customer buy?
- do you have cancer?
- is this spam?
- whose picture is this?
- what is this text about?[j]

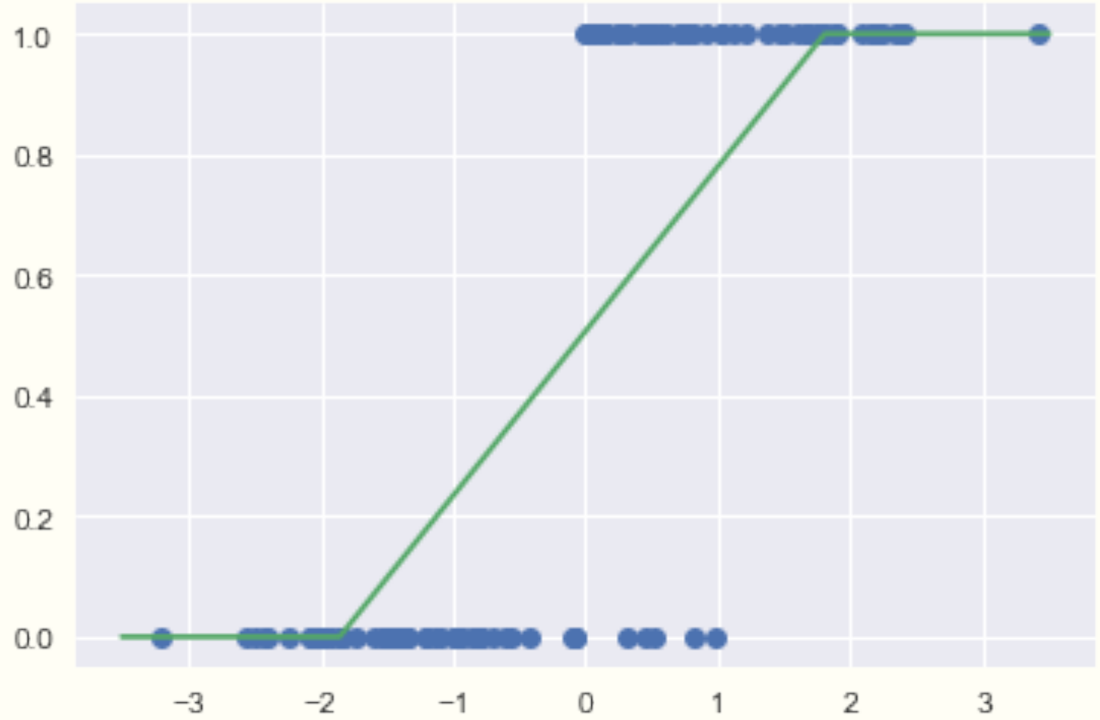
[j] image from code in http://bit.ly/1Azg29G

1-D classification problem



1-D Using Linear regression





2. Logistic Regression

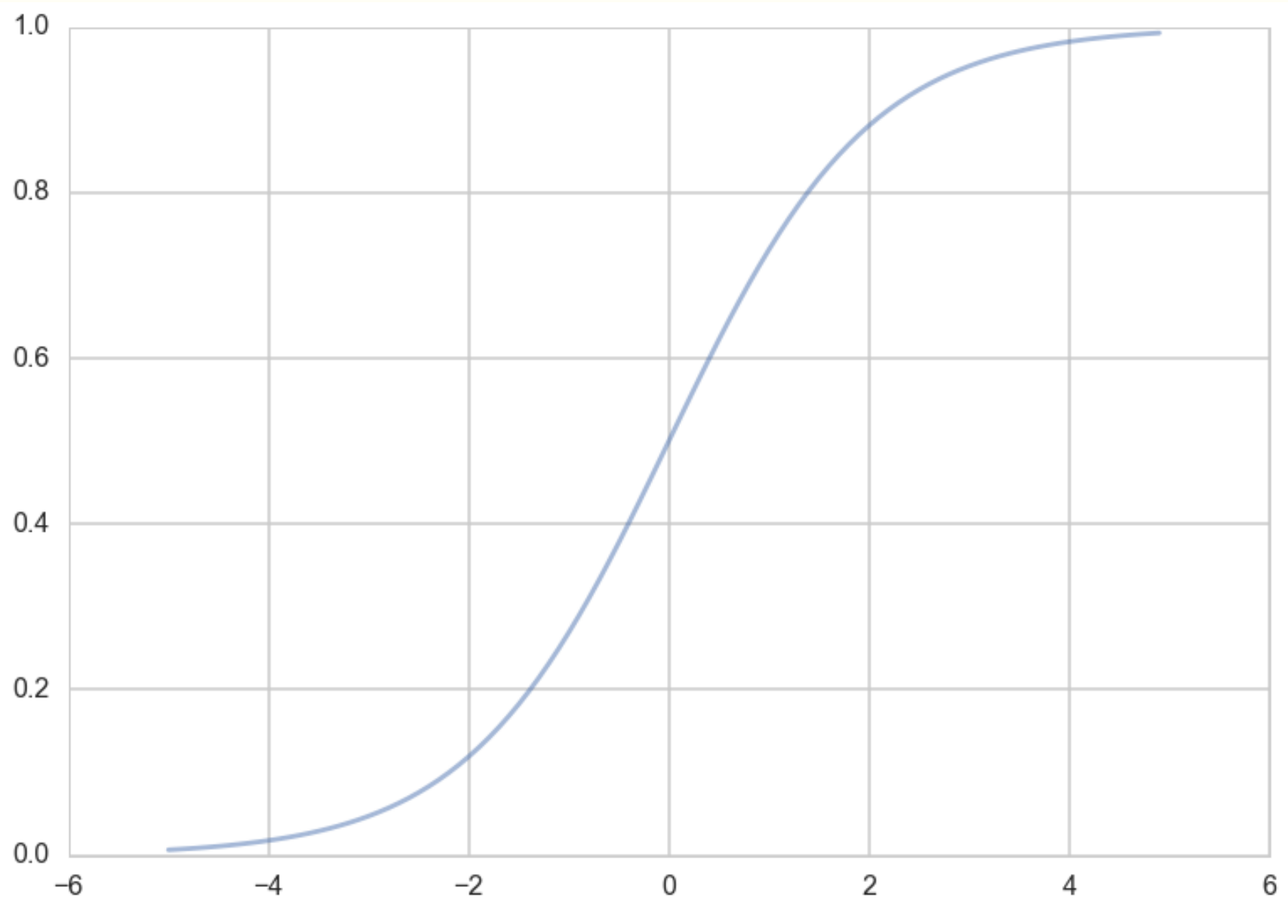
Sigmoid function

This function is plotted below:

```
h = lambda z: 1./(1+np.exp(-z))
zs=np.arange(-5,5,0.1)
plt.plot(zs, h(zs), alpha=0.5);
```

Identify: $z = \mathbf{w} \cdot \mathbf{x}$ and $h(\mathbf{w} \cdot \mathbf{x})$ with the probability that the sample is a '1' ($y = 1$).

In other words, "Squeeze" linear regression through a **Sigmoid** function and this bounds the output to be a probability



Then, the conditional probabilities of $y = 1$ or $y = 0$ given a particular sample's features $\mathbf{x}$ are:

$$P(y = 1|\mathbf{x}) = h(\mathbf{w} \cdot \mathbf{x})$$
$$P(y = 0|\mathbf{x}) = 1 - h(\mathbf{w} \cdot \mathbf{x}).$$

These two can be written together as

$$P(y|\mathbf{x}, \mathbf{w}) = h(\mathbf{w} \cdot \mathbf{x})^y (1 - h(\mathbf{w} \cdot \mathbf{x}))^{(1-y)}$$

This is called a BERNOULLI distribution!!

Each sample is considered independent of the other. Thus, multiplying over the samples we get:

$$P(y|\mathbf{x},\mathbf{w}) = P(\{y_i\}|\{\mathbf{x}_i\},\mathbf{w}) = \prod_{y_i \in \mathcal{D}} P(y_i|\mathbf{x}_i,\mathbf{w}) = \prod_{y_i \in \mathcal{D}} h(\mathbf{w}\cdot\mathbf{x}_i)^{y_i}\left(1-h(\mathbf{w}\cdot\mathbf{x}_i)\right)^{(1-y_i)}$$

Maximum likelihood estimation maximises the **likelihood**, this multiplied probability:

$$\mathcal{L} = P(y \mid \mathbf{x},\mathbf{w}) = P(y|\mathbf{x},\mathbf{w}),$$

OR, alternately the log-likelihood, $\ell = log(P(y \mid \mathbf{x},\mathbf{w}))$.

Thus

$$
\begin{aligned}
\ell &= log\left(\prod_{y_i \in \mathcal{D}} h(\mathbf{w}\cdot\mathbf{x}_i)^{y_i}\left(1-h(\mathbf{w}\cdot\mathbf{x}_i)\right)^{(1-y_i)}\right) \\
&= \sum_{y_i \in \mathcal{D}} log\left(h(\mathbf{w}\cdot\mathbf{x}_i)^{y_i}\left(1-h(\mathbf{w}\cdot\mathbf{x}_i)\right)^{(1-y_i)}\right) \\
&= \sum_{y_i \in \mathcal{D}} log\, h(\mathbf{w}\cdot\mathbf{x}_i)^{y_i} + log\left(1-h(\mathbf{w}\cdot\mathbf{x}_i)\right)^{(1-y_i)} \\
&= \sum_{y_i \in \mathcal{D}} (y_i log(h(\mathbf{w}\cdot\mathbf{x})) + (1-y_i) log(1-h(\mathbf{w}\cdot\mathbf{x})))
\end{aligned}
$$

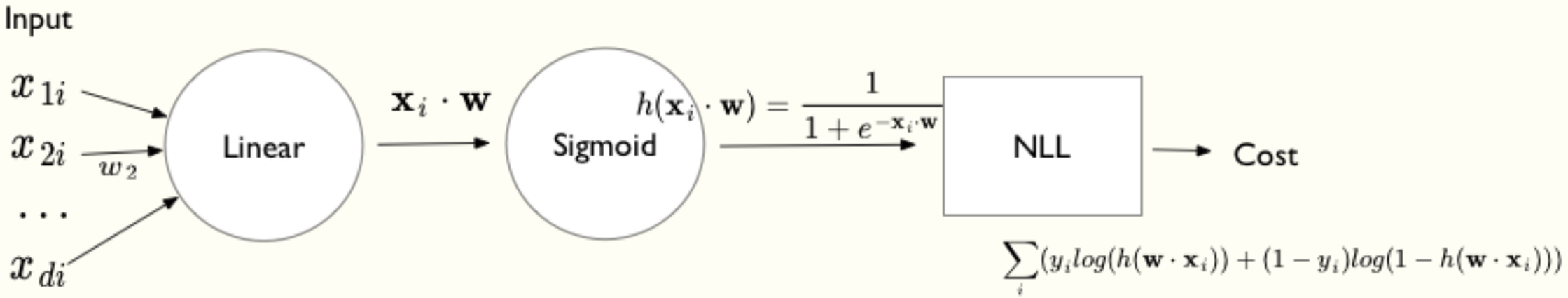
Logistic Regression

The negative of this log likelihood (NLL), also called *cross-entropy*.

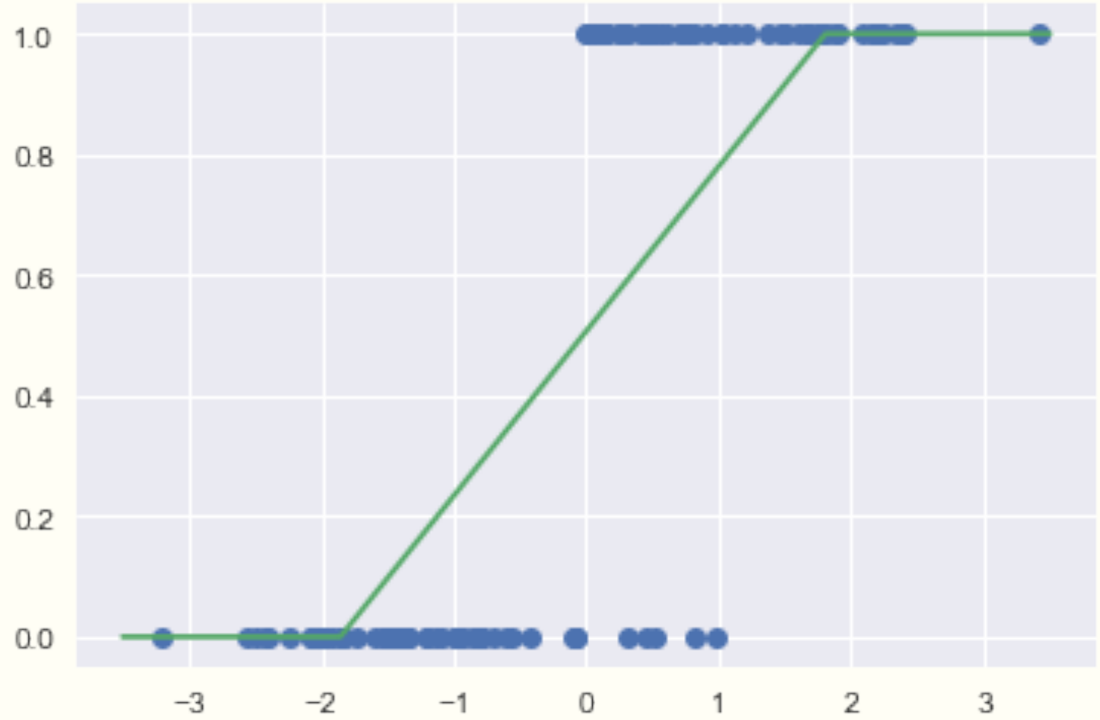
$$NLL = -\sum_{y_i \in \mathcal{D}} (y_i log(h(\mathbf{w} \cdot \mathbf{x})) + (1 - y_i) log(1 - h(\mathbf{w} \cdot \mathbf{x})))$$

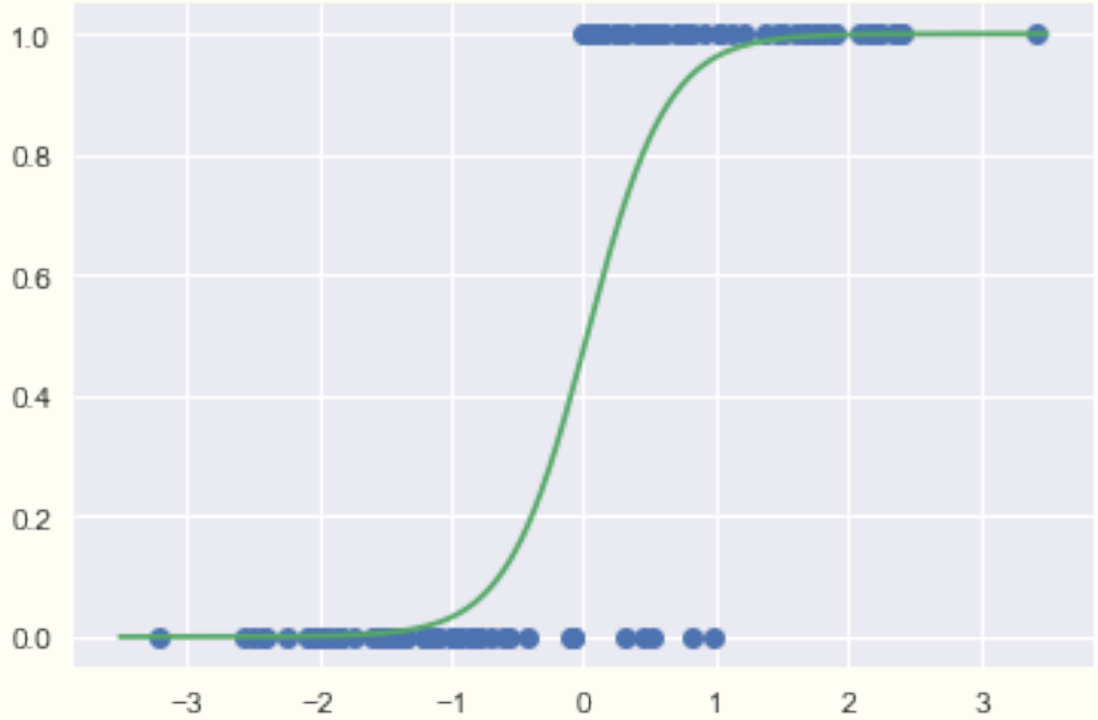
Gradient: $\nabla_{\mathbf{w}} NLL = \sum_i \mathbf{x}_i^T (p_i - y_i) = \mathbf{X}^T \cdot (\mathbf{p} - \mathbf{w})$

Hessian: $H = \mathbf{X}^T diag(p_i(1 - p_i))\mathbf{X}$ positive definite $\implies$ convex


Input
x_{1i}
x_{2i}
w_2
...
x_{di}
Linear
$\mathbf{x}_i \cdot \mathbf{w}$
Sigmoid
$h(\mathbf{x}_i \cdot \mathbf{w}) = \frac{1}{1+e^{-\mathbf{x}_i \cdot \mathbf{w}}}$
NLL
Cost
$\sum_i (y_i log(h(\mathbf{w} \cdot \mathbf{x}_i)) + (1-y_i) log(1-h(\mathbf{w} \cdot \mathbf{x}_i)))$

1-D Using Logistic regression



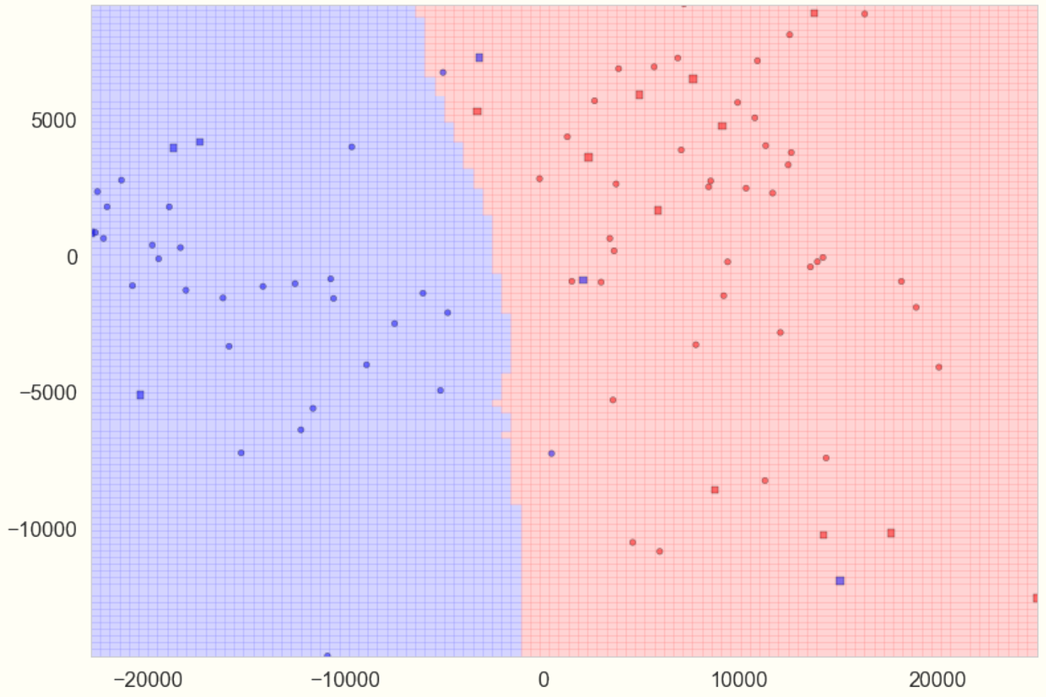


3. Metrics And Risk

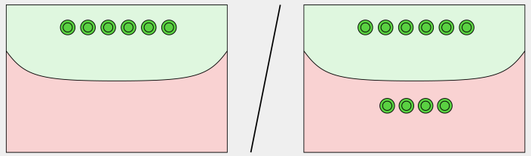
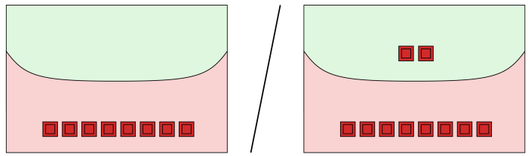
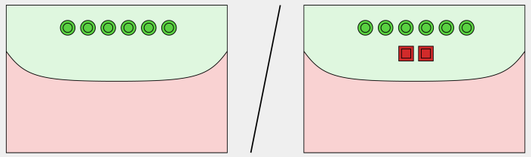
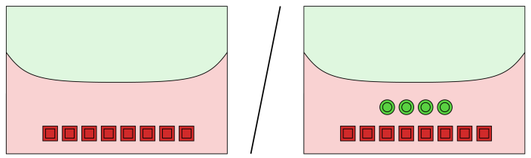
Confusion Matrix

Predicted

		0	1	
Observed	0	TN True Negative	FP False Positive	ON Observed Negative
	1	FN False Negative	TP True Positive	OP Observed Positive
		PN Predicted Negative	PP Predicted Positive	



Simple metrics

Recall	TPR	$\frac{TP}{TP+FN}$	 $= \frac{6}{6+4} = 6/10 = 0.6$
Specificity	TNR	$\frac{TN}{TN+FP}$	 $= \frac{8}{8+2} = 8/10 = 0.8$
Precision	PPV	$\frac{TP}{TP+FP}$	 $= \frac{6}{6+2} = 6/8 = 0.75$
Negative Predictive Value	NPV	$\frac{TN}{TN+FN}$	 $= \frac{8}{8+4} = 8/12 \approx 0.66$

The two risks, or rather risk and score

When we estimate a model using maximum likelihood converted to a risk (how? by NLL) we are calling this risk an **estimation risk**.

Scoring is a different enterprise, where we want to compare different models using their **score** or **decision risk**

The latter leads to the idea of the **Bayes Model**, the best you can do..

Average Classification Risk

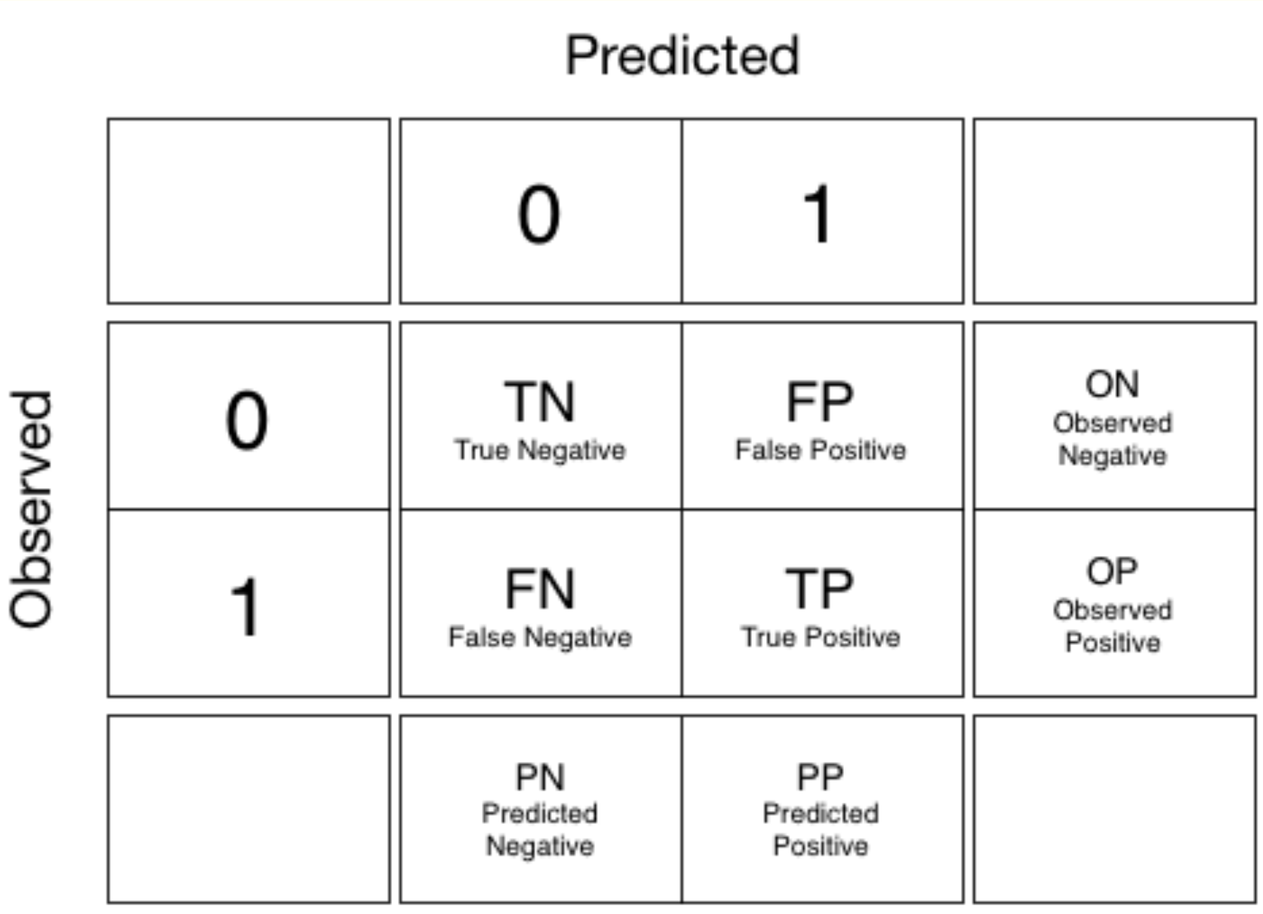
$$R_a(x) = \sum_y l(y, a(x))p(y|x)$$

That is, we calculate the **predictive averaged risk** over all choices y, of making choice a for a given data point.

Overall risk, given all the data points in our set:

$$R(a) = \sum_x R_a(x)$$

Two class Classification



$$R_a(x) = l(1, g)p(1|x) + l(0, g)p(0|x).$$

Then for the "decision" $a = 1$ we have:

$$R_1(x) = l(1, 1)p(1|x) + l(0, 1)p(0|x),$$

and for the "decision" $a = 0$ we have:

$$R_0(x) = l(1, 0)p(1|x) + l(0, 0)p(0|x).$$

CLASSIFICATION RISK

- $R_{g,\mathcal{D}}(x) = P(y_1|x)\ell(g, y_1) + P(y_0|x)\ell(g, y_0)$
- The usual loss is the 1-0 loss $\ell = 1_{g \neq y}$. (over all points $\frac{1}{n}\sum_{i}^{n} I\left(y_i = \hat{y}_i\right)$)
- Thus, $R_{g=y_1}(x) = P(y_0|x)$ and $R_{g=y_0}(x) = P(y_1|x)$

  CHOOSE CLASS WITH LOWEST RISK

1 if $R_1 \leq R_0 \implies$ 1 if $P(0|x) \leq P(1|x)$.

choose 1 if $P(1|x) \geq 0.5$! Intuitive!