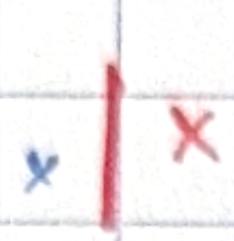
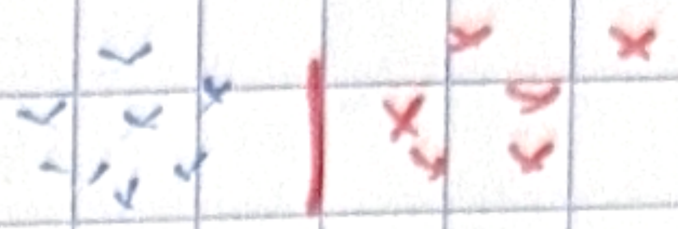
Bias-Variance

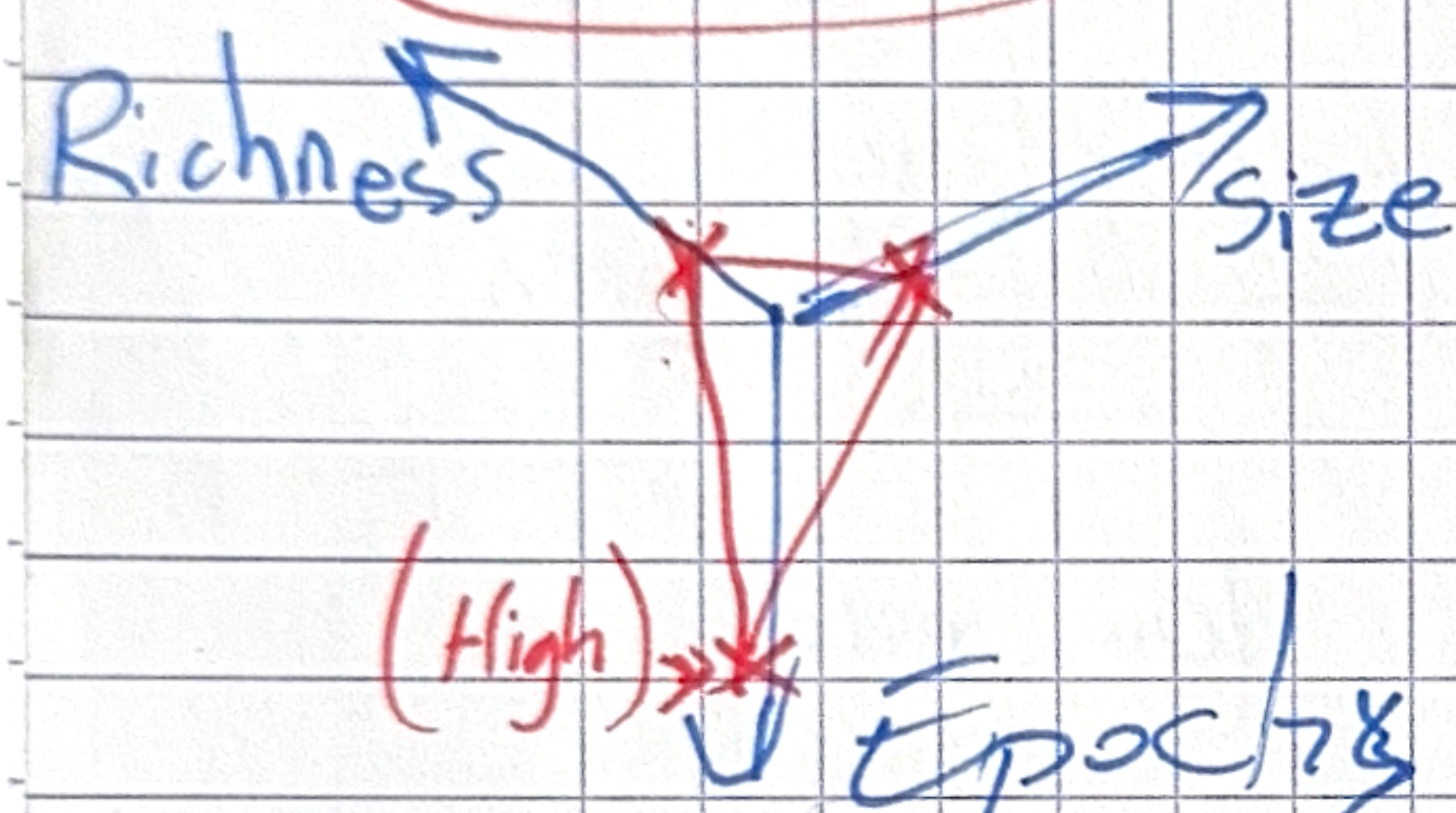
	Bias	Variance
	. underfitting . low feature extraction ↳ misses patterns	. overfitting . over sensitive feature extraction ↳ detects noise $\boxed{\text{catch}}$
Evaluation		
Training data	High error	low error
Text dada	High error	High erros
Models		
	. Linear Regression . Naive Bayes	. Deep Neural Network . Decision Trees (no prune)
Causes		
	. low model complexity	. High model complexity
Generalisation	↳ poor	↳ poor

Overfitting

Why

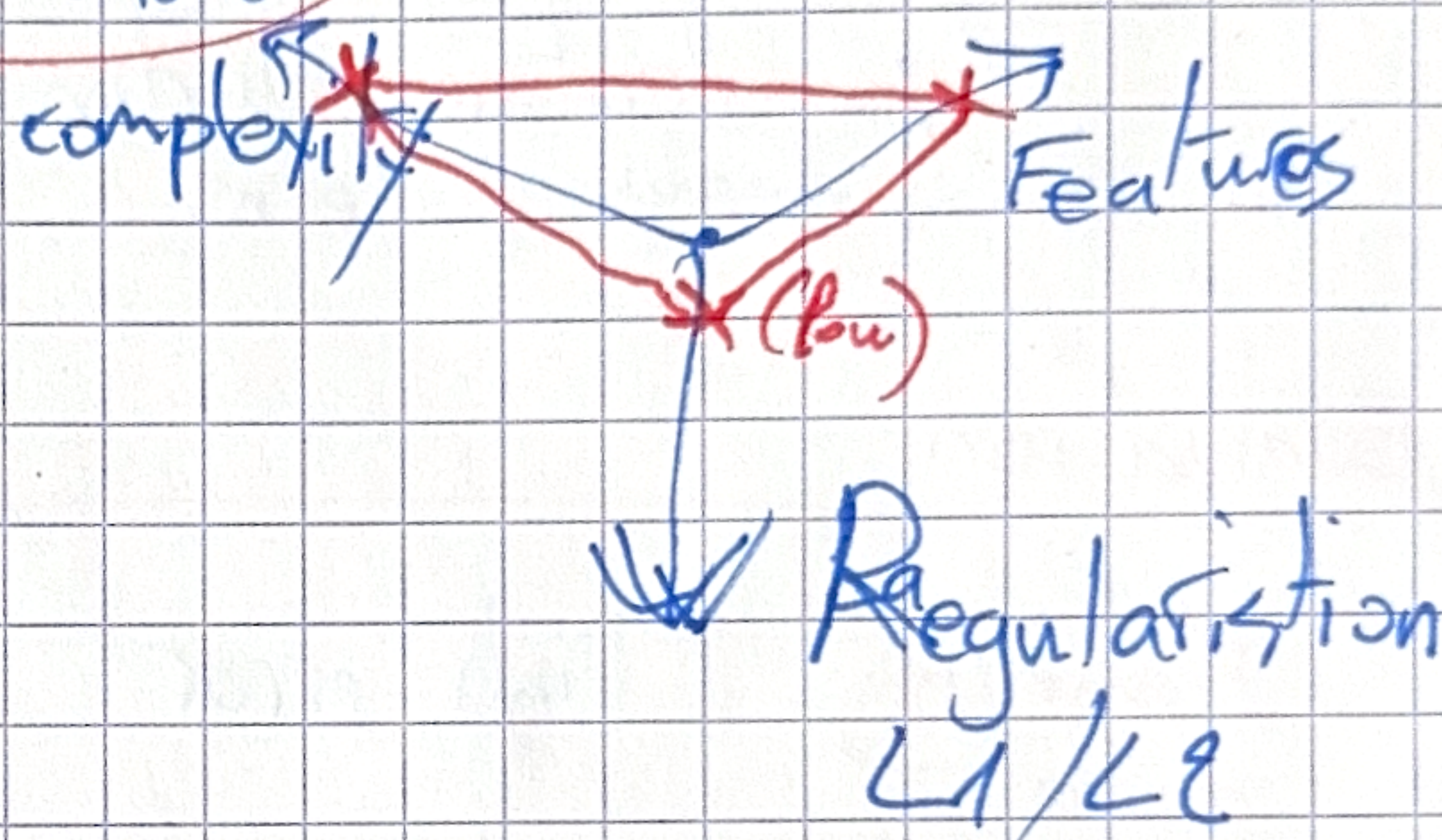
External

Dataset



~~External~~ Internal

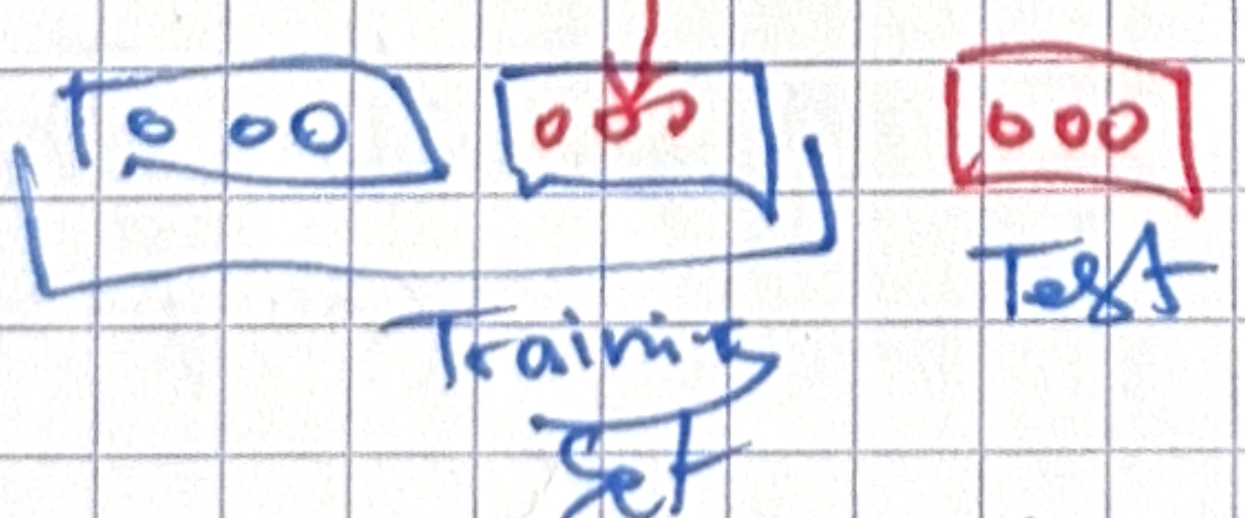
Model



Solutions

Dataset

- Increase size
- Data augmentation
- Cross validation



- Reduce training time / epochs.

Model

- Reduce complextiy (layers number)
- Feature selection
- Dropouts (NN)
  ↳ Random neurons.

Precision / Recall / F1

Model 1

Model 2

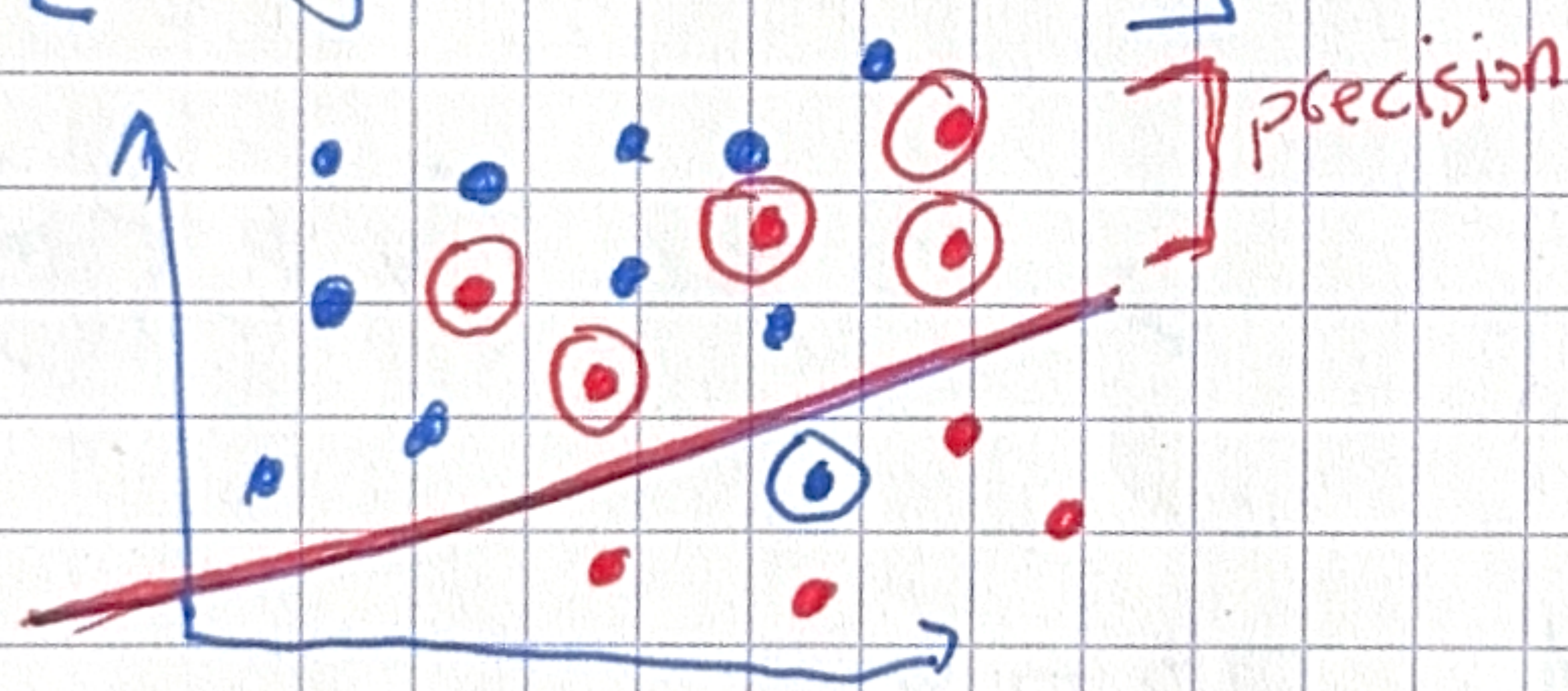
patients (Actual)

diagnosis (Prediction)

	S	H
Truely S	10	1
Truly H	6	4

True positive
True Negative
False positive
False negative

[Confusion Matrix]



Accuracy: $\frac{10+4}{10+1+4+6} = \frac{14}{21} = 0.66$ $\boxed{\frac{TP+TN}{TP+TN+FP+FN}}$

↳ $\boxed{66\%}$ of the classifications were correct

Precision: $\frac{10}{10+6} = \frac{10}{16} = 0.62$ $\boxed{\frac{TP}{TP+FP}}$

↳ $\boxed{62\%}$ of patients classified as sick, are actually sick

Recall: $\frac{10}{10+1} = \frac{10}{11} = 0.91$ $\boxed{\frac{TP}{TP+FN}}$

↳ $\boxed{91\%}$ of actual sick patients were detected

F1: $\frac{2 \times 62 \times 91}{62+91} = \frac{11284}{153} = \boxed{73.7\%}$ $\boxed{\frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}}$

F1 / F_β

↳ Harmonic Mean between precision & Recall

Math

↳ Arithmic Mean $\frac{x+y}{2}$

x × ——— * ——— × y

↳ Harmonic Mean

× ——— * ——— × y

The least will bring down the score. ⟵ $\frac{2 \times x \times y}{x+y}$

F_β

Precision ——— F1 ——— Recall

(ex: 0.5) $\beta < 1$ ⟵

⟶ $\beta > 1$ (ex 2, 5, 10)

$$F_1 = \frac{2 \times 62 \times 91}{62 + 91} = 73.7\%$$

$$F_{0.5} = (1 + 0.5^2) \times \frac{62 \times 91}{(0.5^2 \times 62) + 91}$$

$$F_\beta = (1 + \beta^2) \times \frac{\text{Precision} \times \text{Recall}}{(\beta^2 \times \text{Preci}) + \text{Recall}}$$

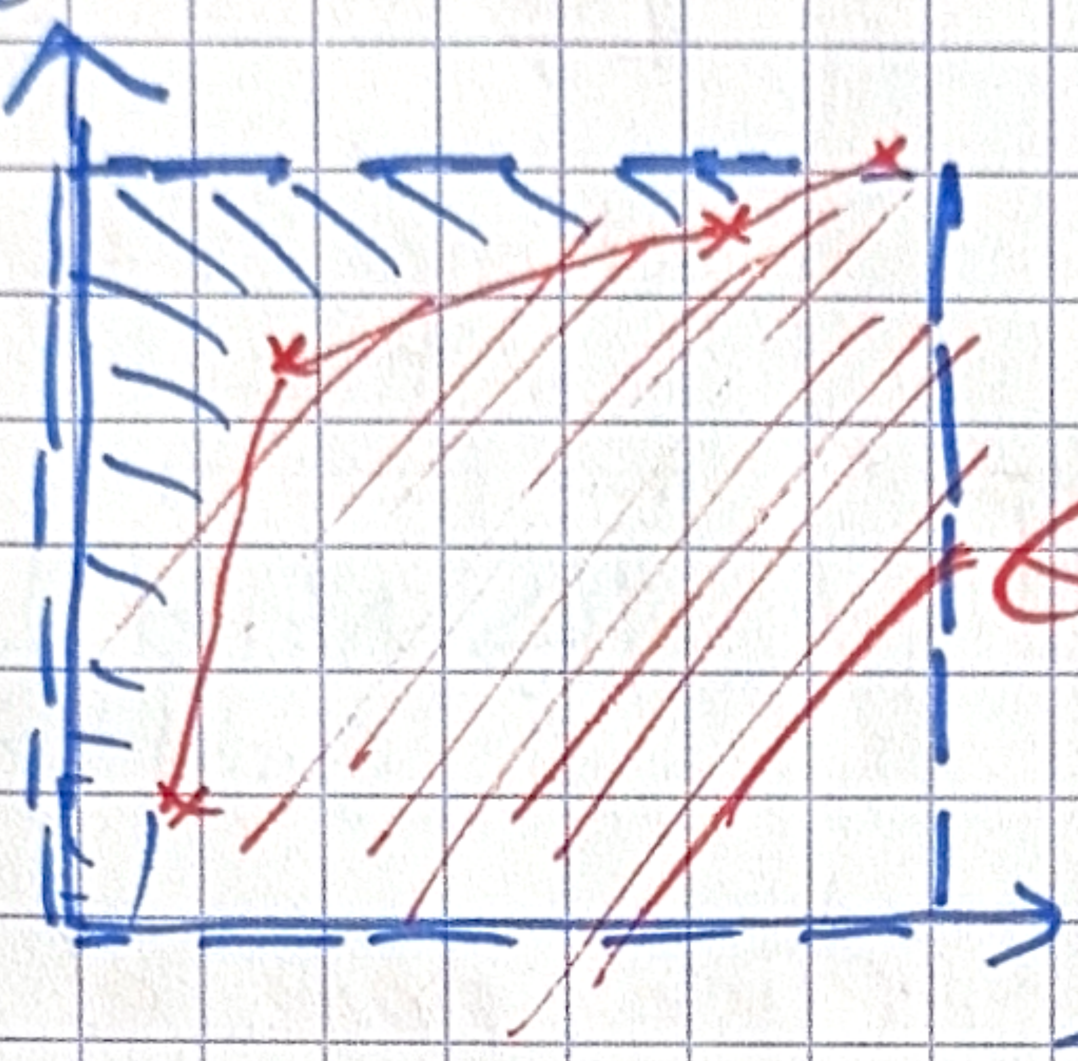
$$\boxed{F_{0.5} = 66{,}22\%}$$

$$F_5 = (1 + 5^2) \times \frac{62 \times 91}{(5^2 \times 62) + 91}$$

$$\boxed{F_5 = 89{,}39\%}$$

ROC - AUC

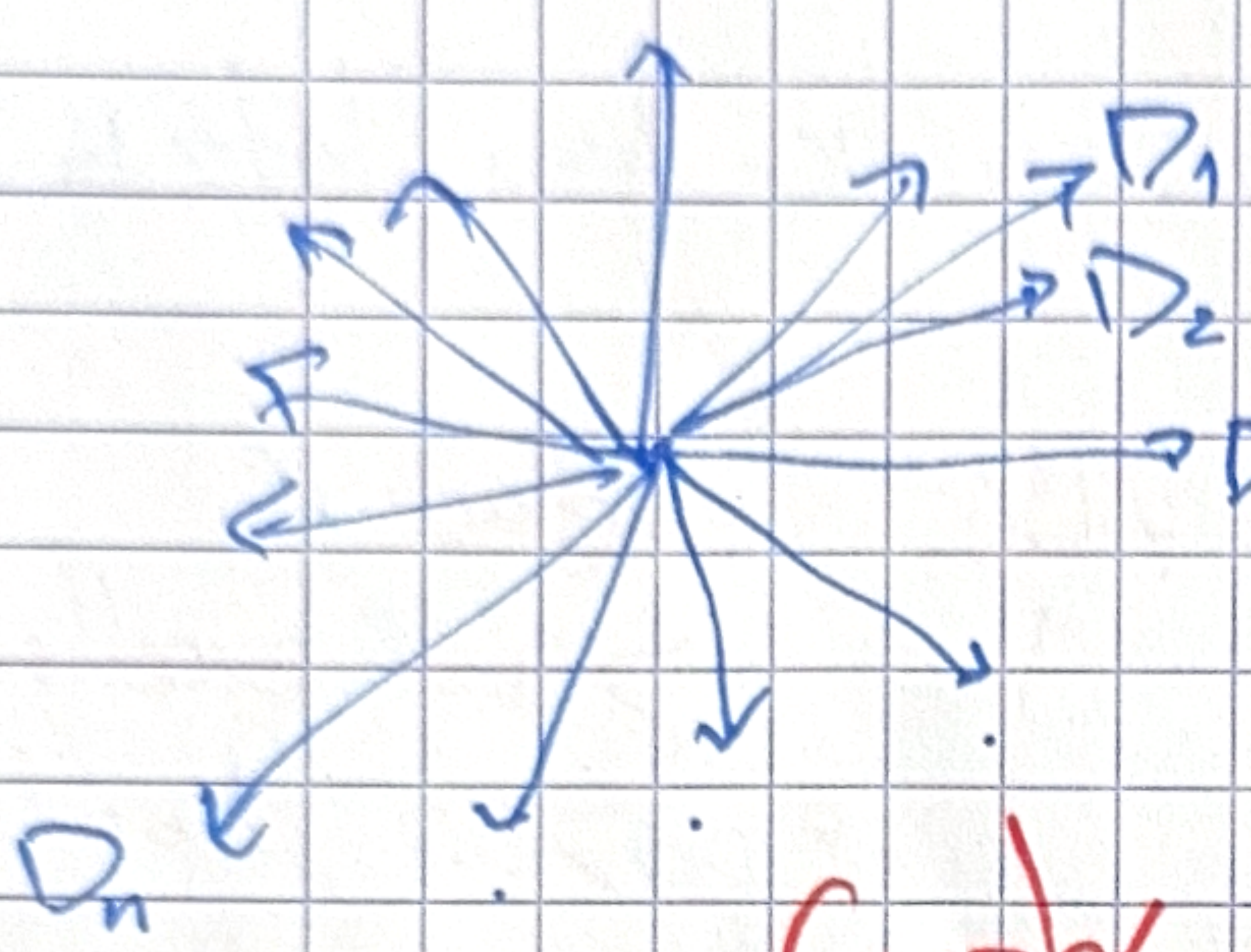
↳ Receiver Opearting Characteristic - Area Under the Curve



AUC surface s

- s > 9.0 excellent
- 0.8 > s > 9.0 Good
- 0.7 > s > 0.8 Acceptable
- 0.6 > s > 0.7 Poor
- s < 0.5 Useless

The curse of high dimensionality



Dimension ↗ 1

↳ Number of features ↗

Hard to extract patterns ① ②

↳ Need to more data for training

Complex model ⇐

⇓

Overfitting ⇐ Captures noise as feature

- Why it's harder to extract patterns?
  - ↳ high dimensions causes the high sparsing of data point
    - ↳ ~~This~~ if no groups are found no feature is selected
      - ↳ So we need more data to hope find an emerging group

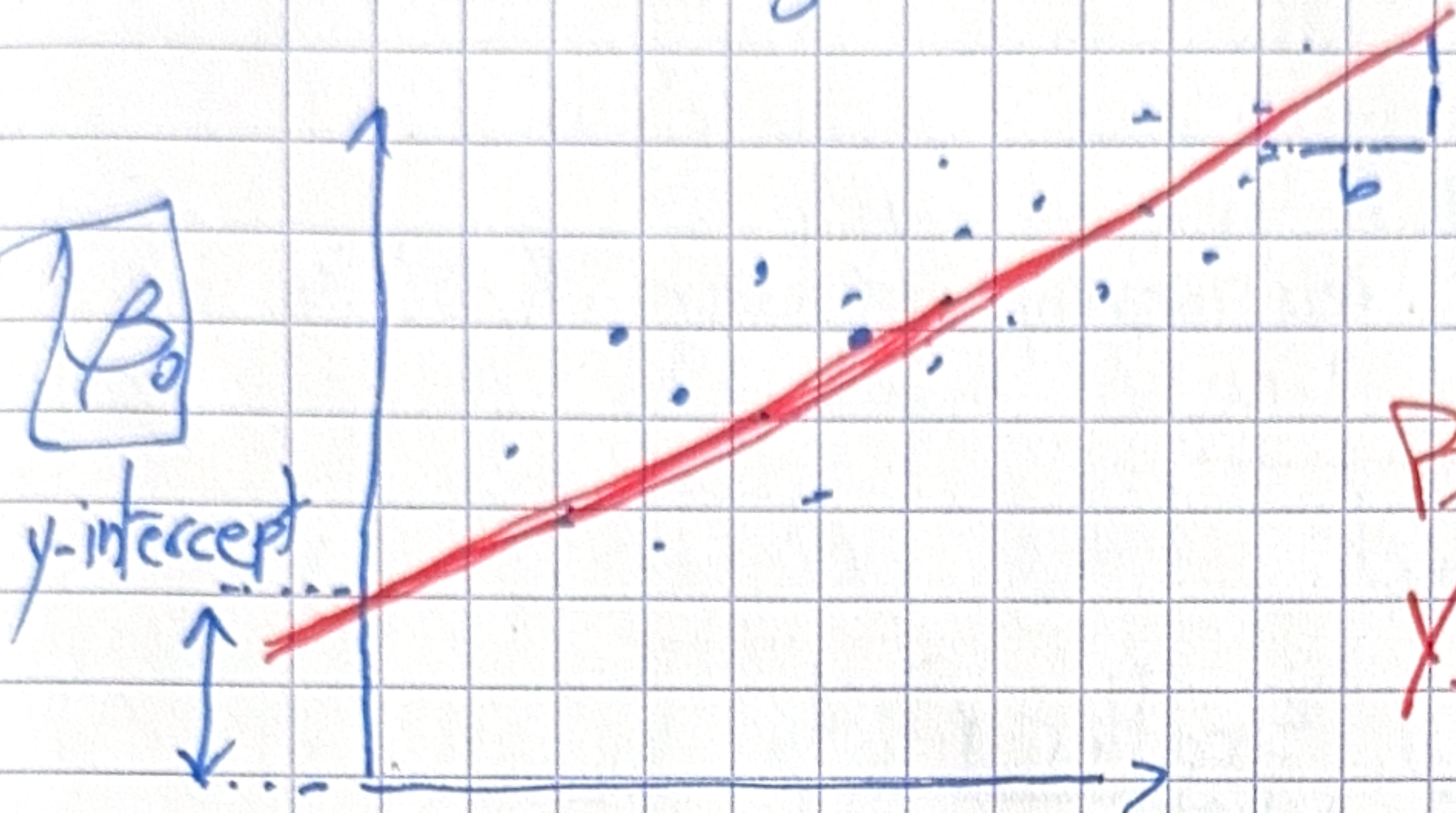
Dimensions ↗ 2

- Caused by
  - dataset complexity (text, images, audio...)

Mitigation

- Manual/Automated feature selection
- Dimensionality reduction
  ↳ PCA, Autoencoder
- Regularisation
  ↳ L1, L2, dropout
- More data

Linear Regression

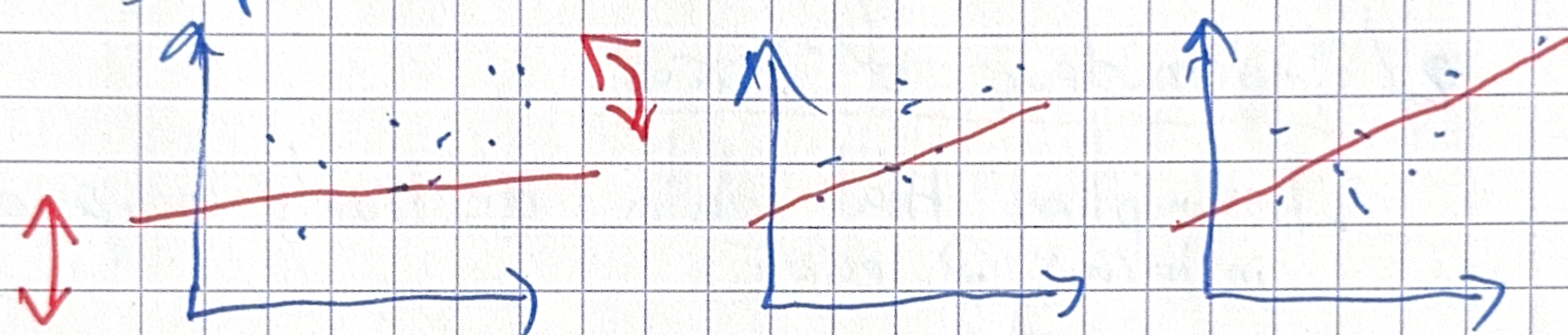


Slope $\frac{a}{b} = \boxed{\beta_1}$

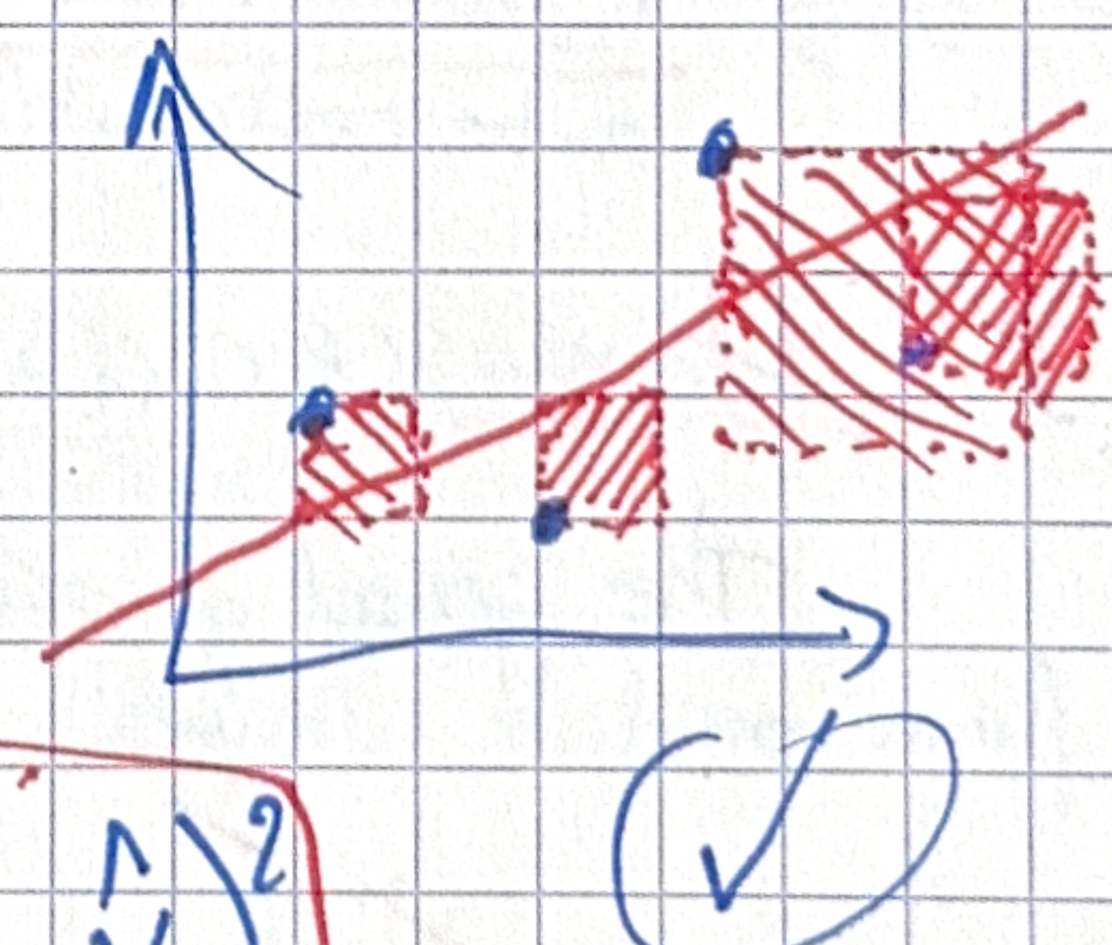
Best-fitting line (Regression line)

$Y = \beta_0 + \beta_1 X + \epsilon$ (error)

Steps



Best-fitting line = Best Mean Square Error (least)



$$MSE\ \cancel{SME} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

Y_i : real y

$\hat{Y}_i$: predicted.

Assumptions of ~~linearity~~ Regression

1/ Linearity

- Assumption of realionship beteeen X, Y is a straight line
- How to check
  - ↳ Plotting Scatter Plots
  - ↳ Residual Plots → TODO

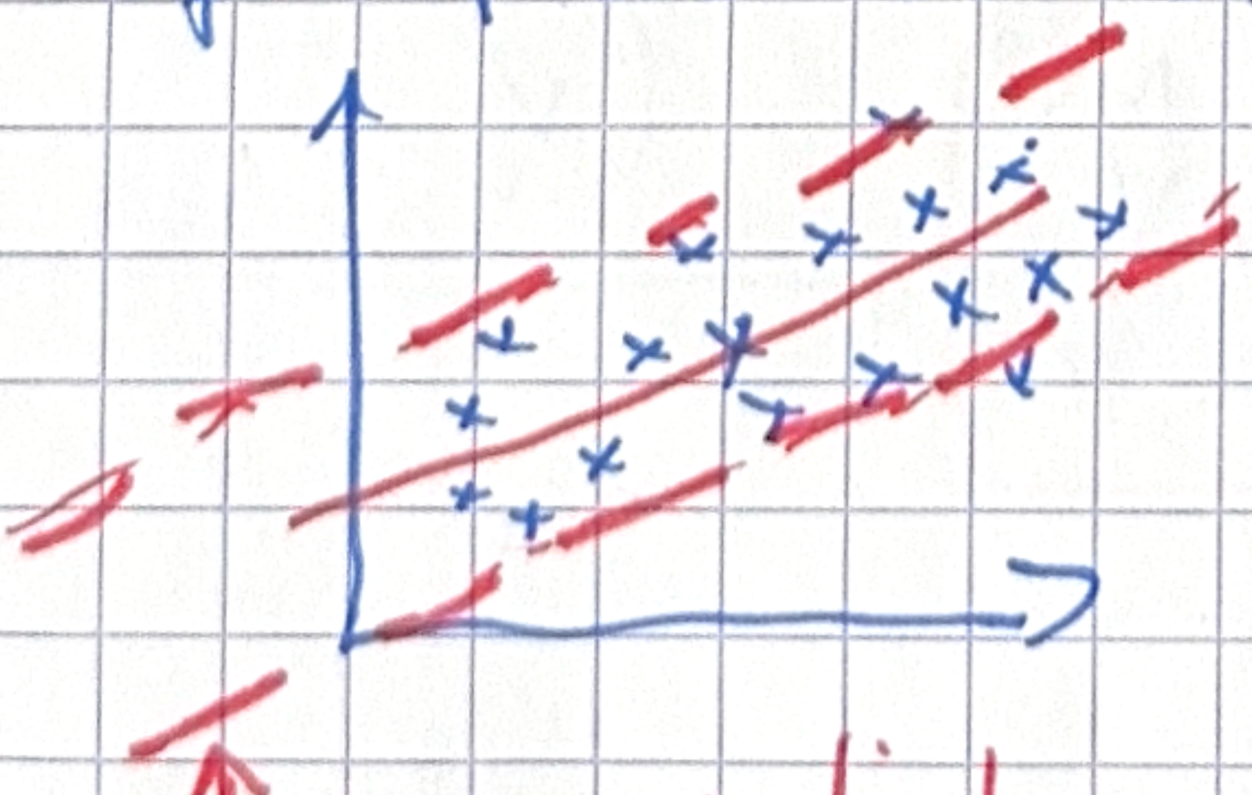
2/ Independence of Error

- Assumption that data are truely independent in terms of error
- How to ckeck
  - ↳ Contextual Domain Knowledge (check for correlation)
  - TODO ← ↳ Durbin–Watson Test (Check for autocorrelation in residuals

3/ Homoscedasticity

The spread of errors is roughly the same from predicts values.

residual == error



Homoscedasticity

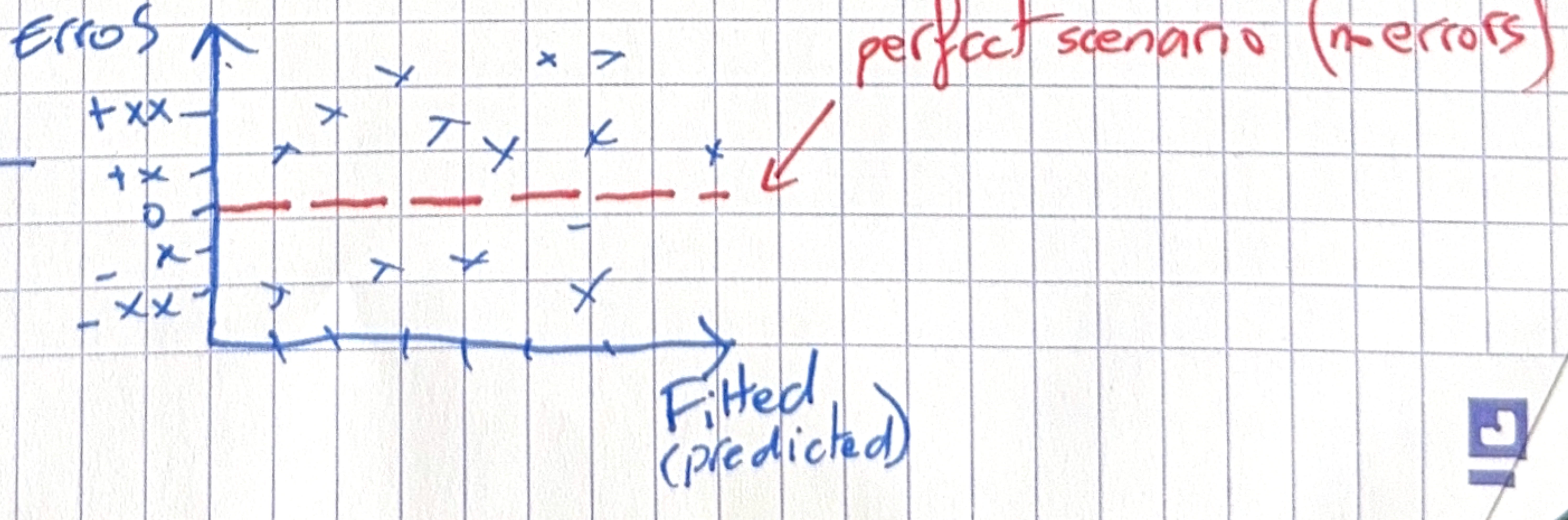
Heterosceadacticity

4/ Normality of Errors

- Errors (Residuals) to be normally distributed (randomly scattered) around zero
  ↳ No obvious patterns

How to check for
↳ Q-Q plot → TODO

How to check for Homoscedasticity
↳ Residuals vs Fitted Plot



Assumptions of linear Regression

1/ Linearity

- Assumption of realationship beteeen X, Y is a straight line
- How to check
  - ↳ Plotting Scatter Plots
  - ↳ Residual Plots → TODO

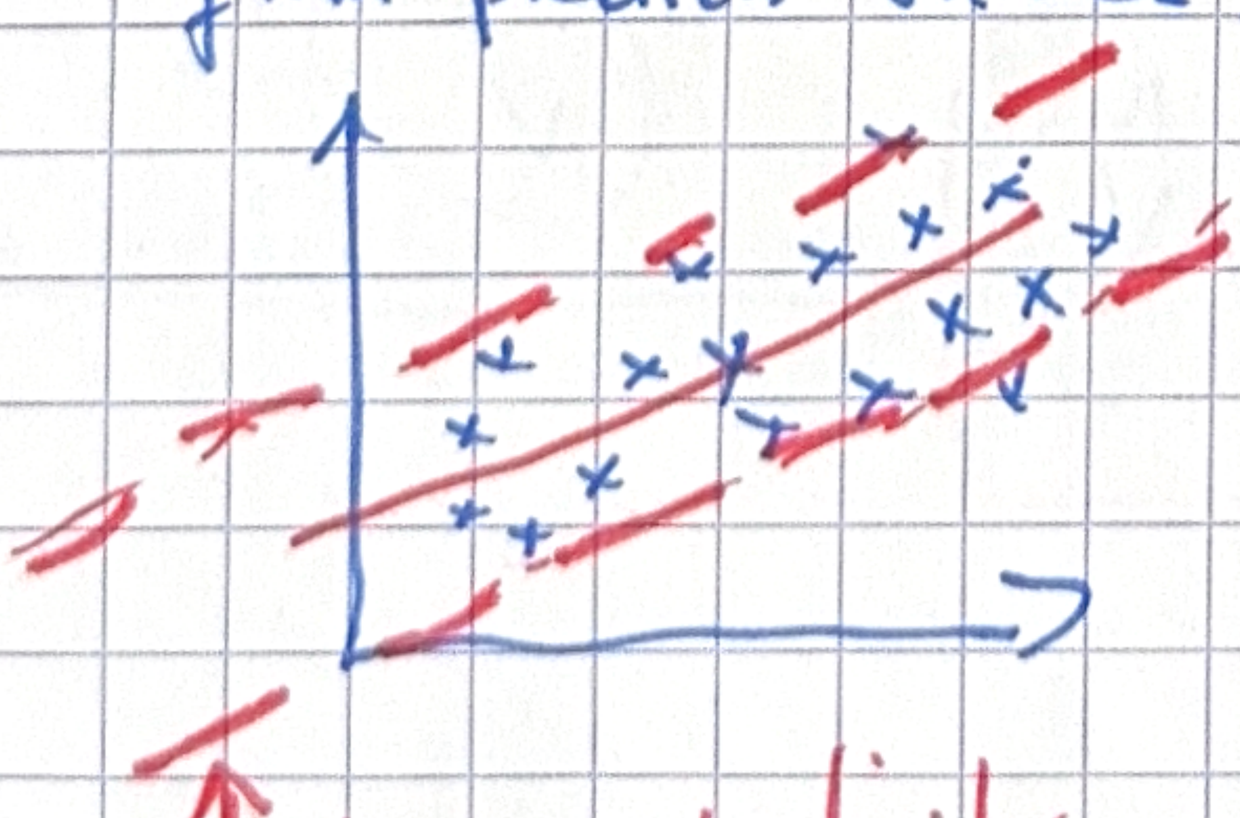
2/ Independence of Error

- Assumption that data are truely independent in terms of error
- How to ckeck
  - ↳ Contextual Domain Knowledge (ckeck for correlation)
  - TODO ← ↳ Durbin–Watson Test (Check for auto correlation in residuals

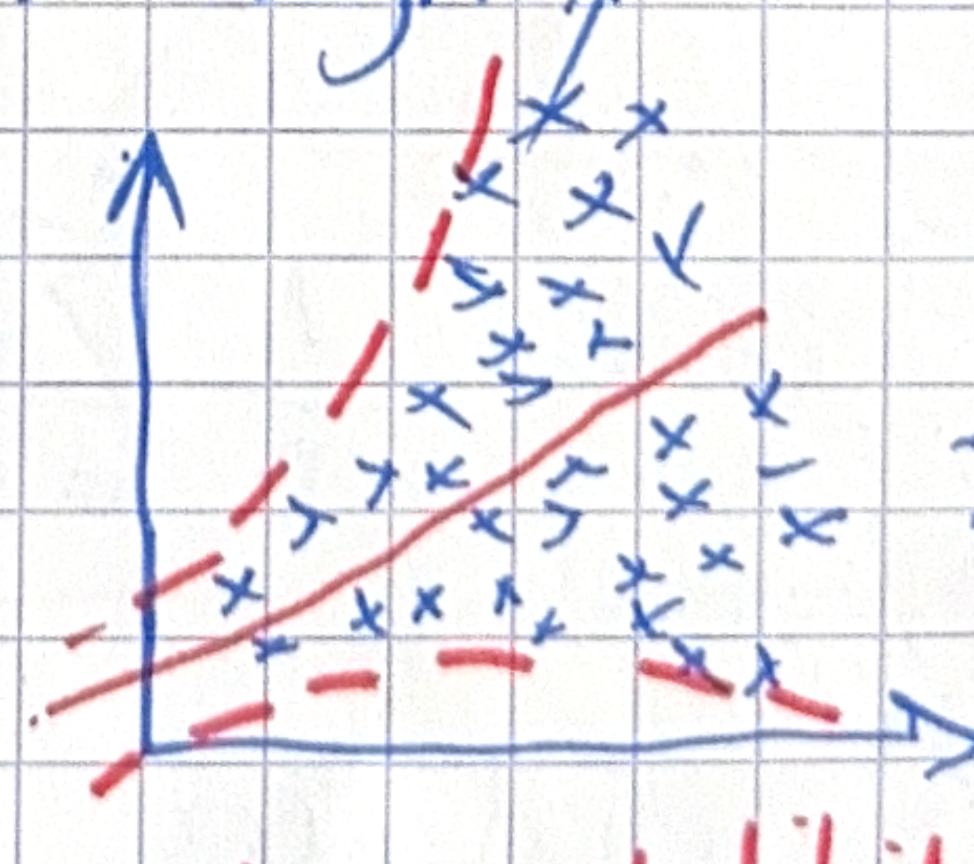
3/ Homosceclasticity

The spread of errors is roughly the same from predicts values.

residual == error



Homoscedasticity



Heteroscedacticity

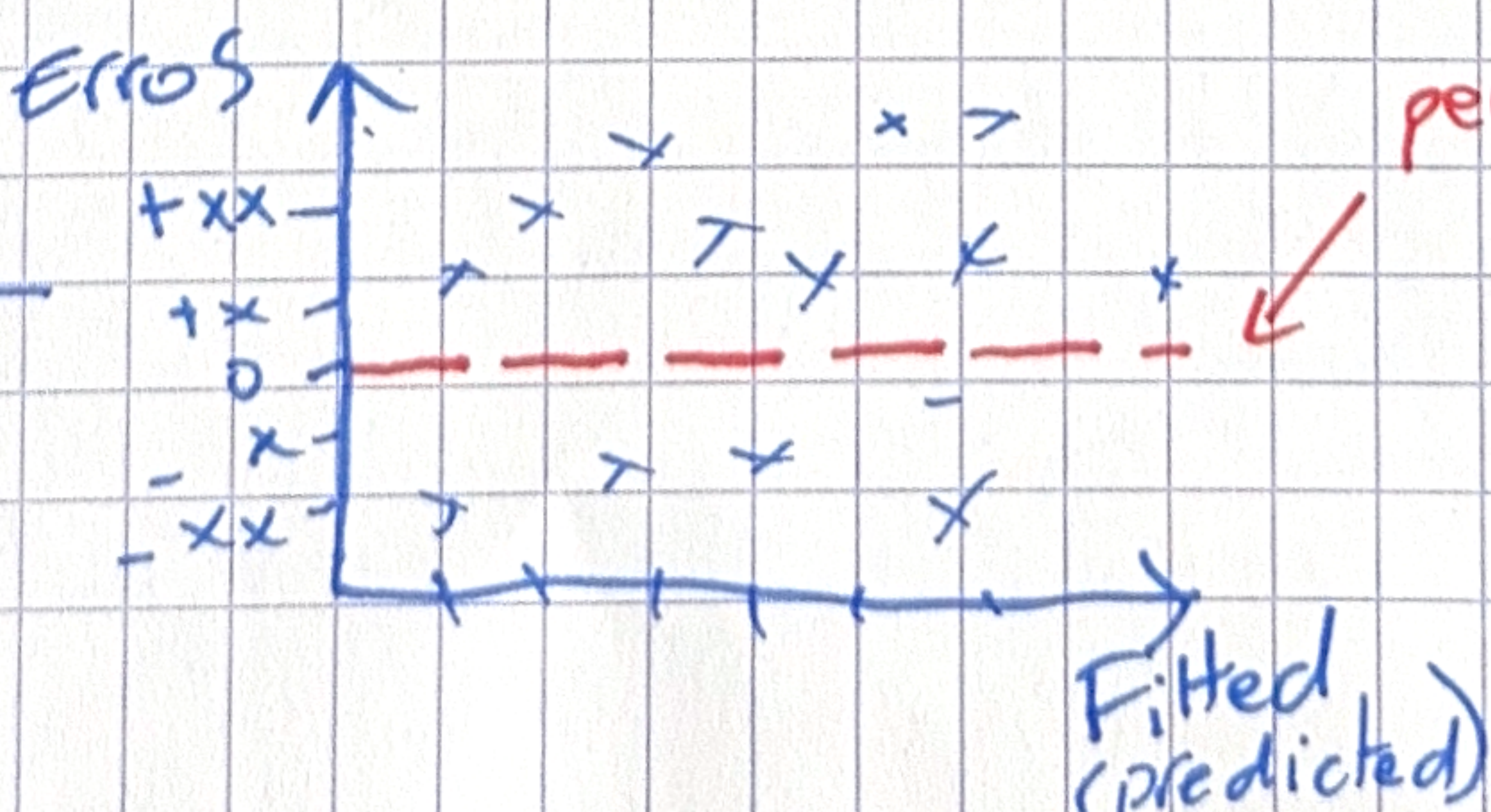
4/ Normality of Errors

- Errors (Residuals) to be normally distributed (randomly scattered) around zero
  ↳ No obvious patterns

How to check for
↳ Q-Q plot → TODO

How to check for Homoscedastisity
↳ Risiduals vs Fitted Plot

perfect scenario (no errors)



5/ No (or Low) Multicollinearity

Multicollinearity: one or more predictor variable are strongly correlated,

⟹ Hard for the regression model to ~~predict~~ quantify the impact of each one of them on prediction

Ex:
- x_1 : house ~~price~~ size
- x_2 : Number of rooms

high correlation

- x_3 : Age of house

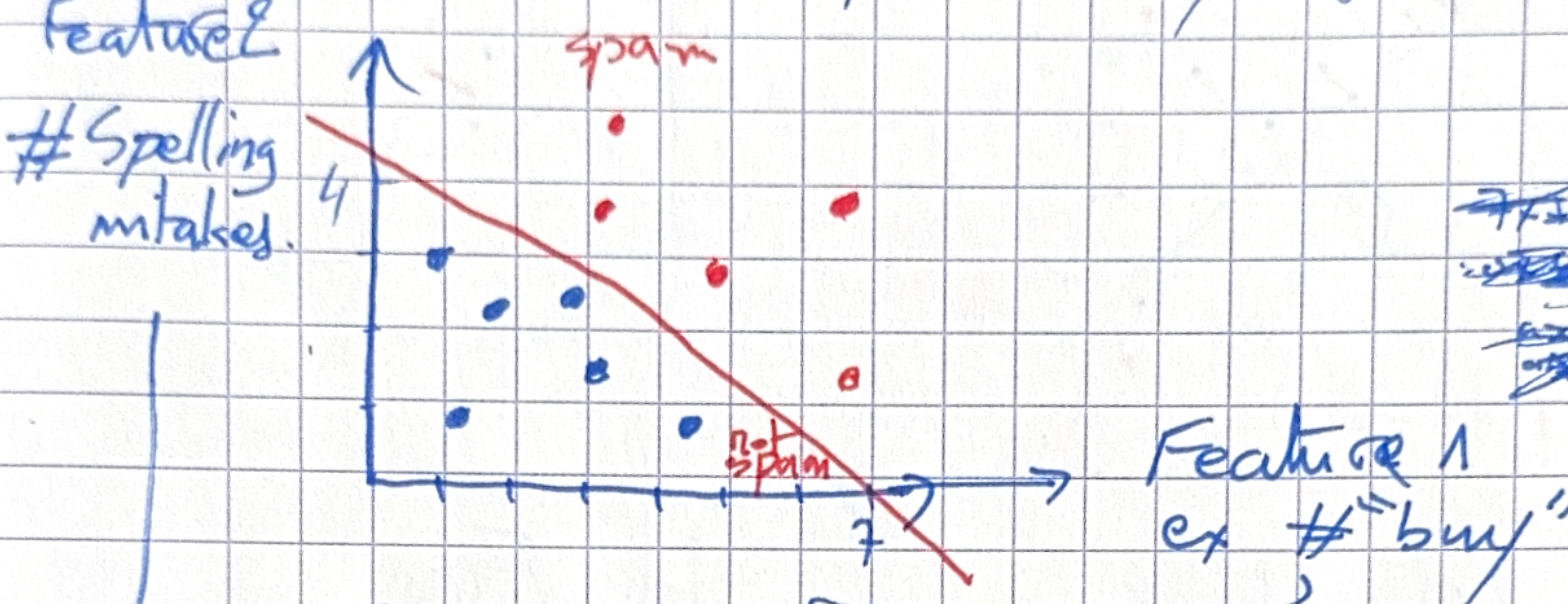
↳ Lead to over-fitting

Solution

↳ drop (or combine) features.

Logistic Regression

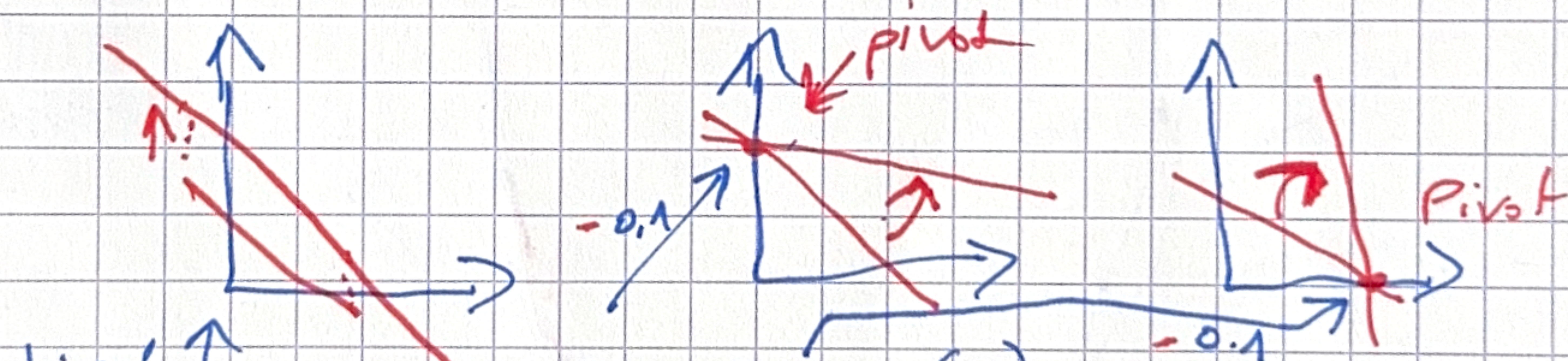
↳ used for 0/1 prediction
↳ also the probability of (0 or 1)



$$\text{Score} = w_1 + w_2 + b$$

$$2x + 3y = 6 \; / \; 2x + 3y - 6 = 0$$

Line movement:



$$2x + 3y + (-6) = 0$$

-0.1

$$1.9x + 2.9y + (-6.1) = 0$$

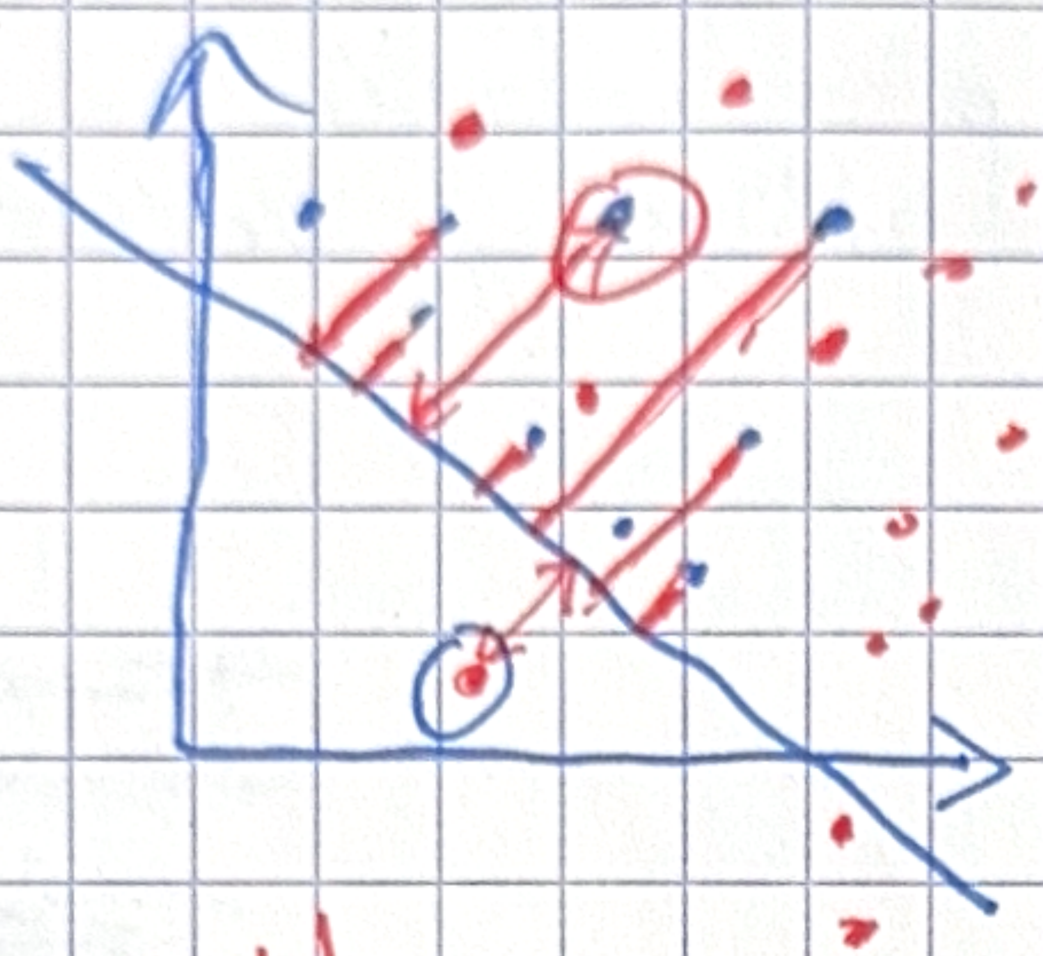
perceptron trick

$$2x + 3y + (-6) = 0$$

$$+ \; (4x - 0.1) + (5x - 0.1) + (-0.1)$$

$$1.96x + 2.95 + (-6.01) = 0$$

Perceptron Error in logistic Regression



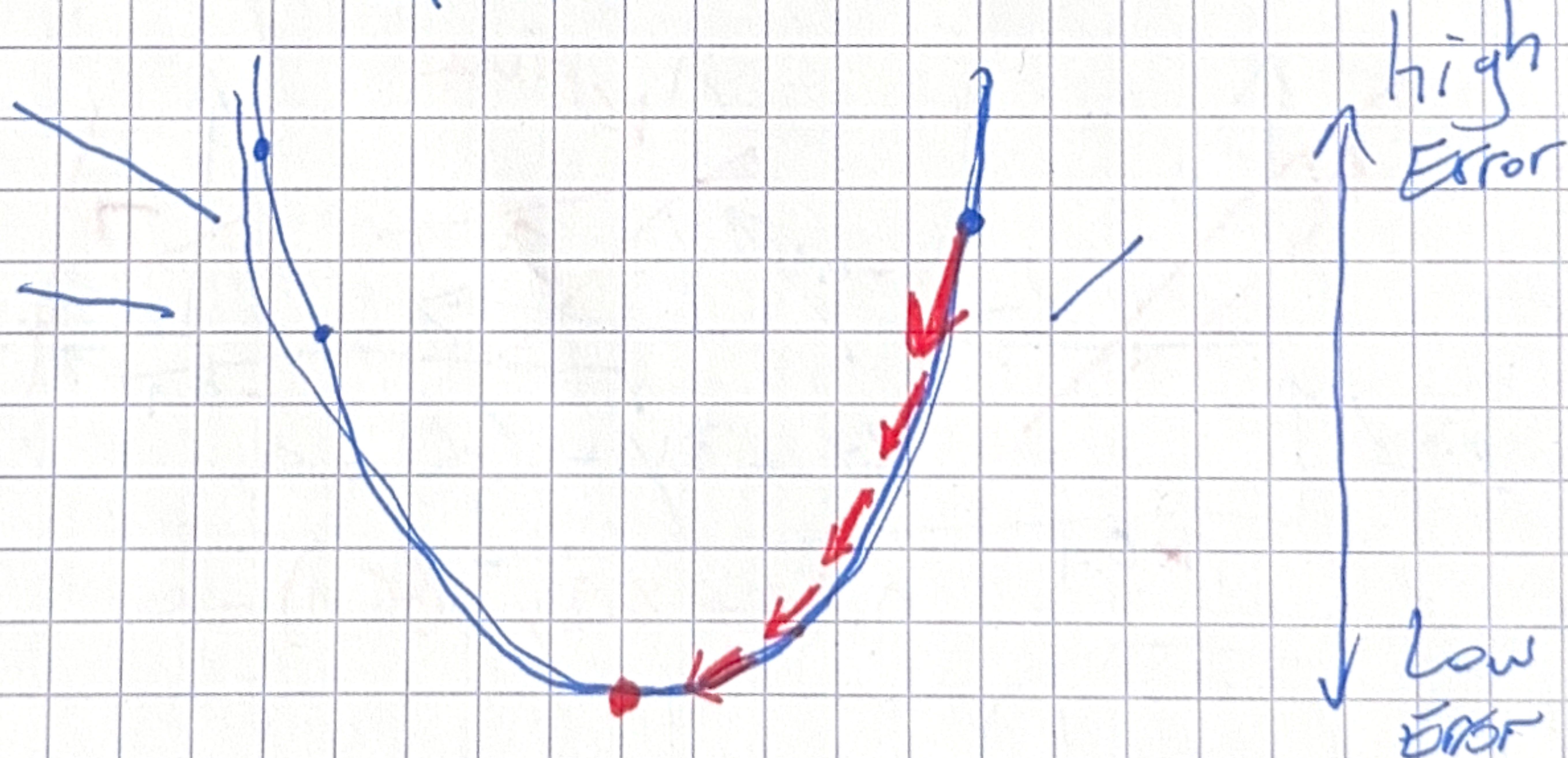
Large Errors ⟶ less Errors

Minimisation Case ⇒ Could by done by calculs (

⇓

Gradient Descent

↳



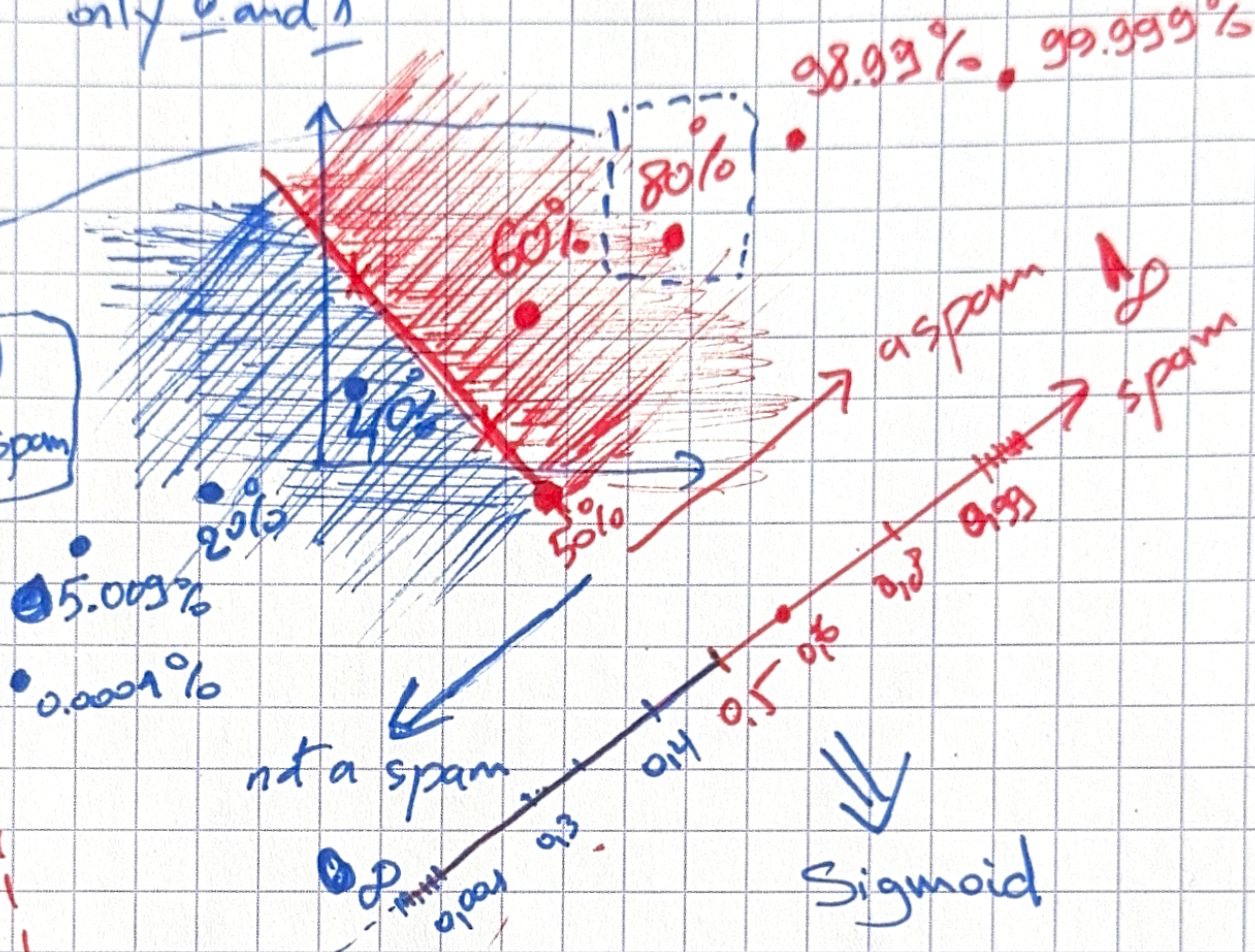
↳ Or use Log-Loss Error ⟶ TODO

To understand Log-Less Error (Error function)

↳ think in Continous Predictions intead of only 0 and 1

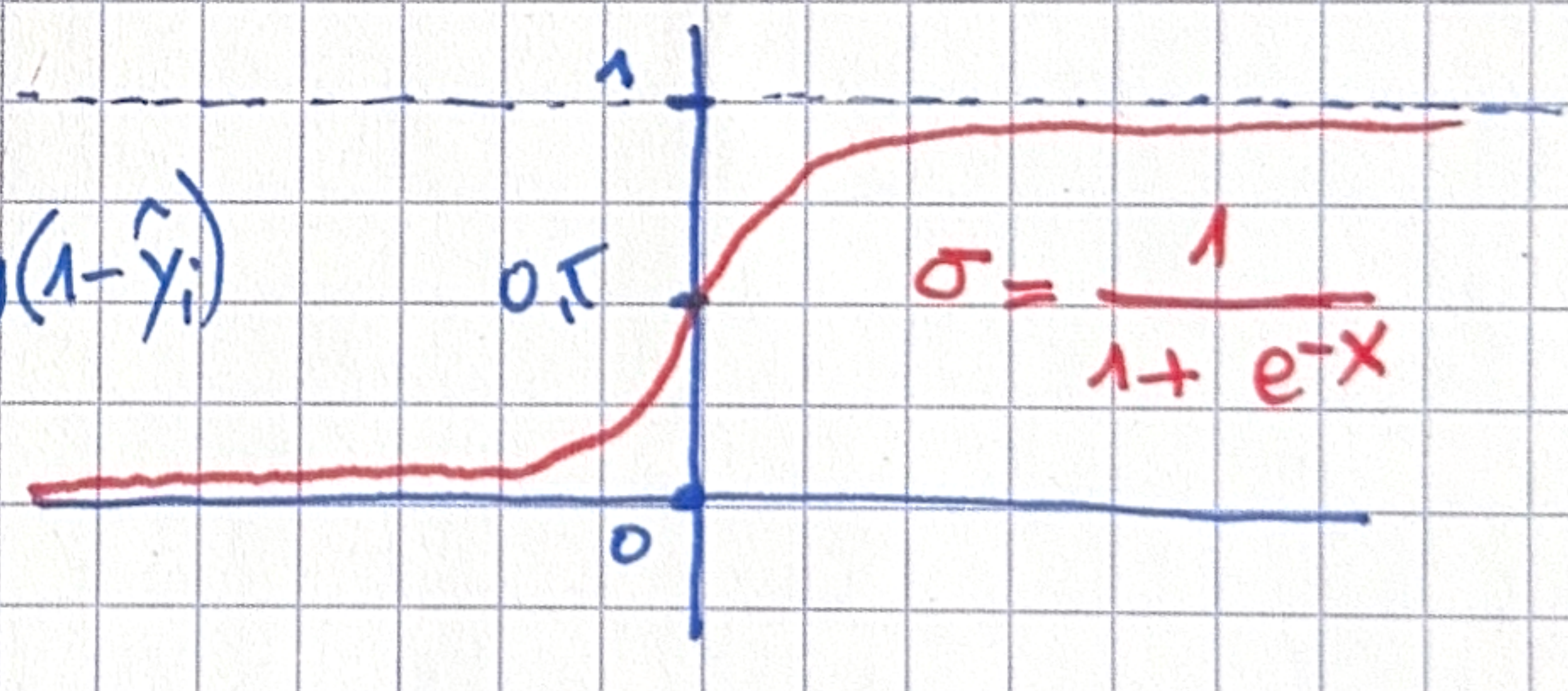
$y = 1$ (spam)

$\hat{y} = 0.8$ likly spam



⇓

Sigmoid



Log-loss =

$$-\frac{1}{n}\sum_{i=1}^{n}\left[y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)\right]$$

⇓ Simplify using Derivatives

$$D = -\frac{1}{n}(y_i - \hat{y}_i)x_i$$