Linear and Quadratic Discriminant Analysis (LDA, QDA)

Joint, marginal, and conditional probability

Example: The following matrix lists the probabilities of having disease X and symptoms Y.

	X = 0	X = 1
Y = 0	0.5	0.1
Y = 1	0.1	0.3

Joint probability = $\mathrm{P}(\mathrm{X} = 1 \cap Y = 1)$ = $\mathrm{P}(\mathrm{X} = 1, Y = 1)$ = 0.3

German literature English literature

probability of 2 event simultaneously happen

Sum over all joint probabilities is: $\sum_{x,y} P(X = x, Y = y) = 1$

The so-called **marginal** probability implies that we keep one of the variables fixed:

$$P(X = x) = \sum_{y} P(X = x, Y = y) \qquad P(Y = y) = \sum_{x} P(X = x, Y = y)$$

Examples: $P(X = 0) = 0.5 + 0.1 = 0.6$ $P(Y = 1) = 0.1 + 0.3 = 0.4$

Conditional probability: $P(X = x|Y = y) = \frac{P(X = x, Y = y)}{P(Y = y)}$ Example: $P(X = 1|Y = 1) = \frac{0.3}{0.1+0.3} = \frac{3}{4}$

The covariance matrix

The figure shows sample points of a multivariate normal distribution with

mean $\mu = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$ and std $\Sigma = \begin{pmatrix} 1 & 3/5 \\ 3/5 & 2 \end{pmatrix}$

shown along with the 3-sigma ellipse, the two marginal distributions, and the two 1D histograms.

The covariance matrix is defined as:

$$\Sigma_{i,j} = E\left[(X_i - \mu_i)(X_j - \mu_j)\right] = Cov\left[X_i, X_j\right]$$

The probability density function is defined as:

$$f(x) = \frac{1}{(2\pi)^{p/2}|\boldsymbol{\Sigma}|^{1/2}} e^{-\frac{1}{2}(x-\mu)^T \boldsymbol{\Sigma}^{-1}(x-\mu)}$$

Example:

The variance of X_2 is twice as high as for X_1. The covariance matrix describes how each variable in your data moves together with every other variable, illustrated by the tilt of the ellipse.

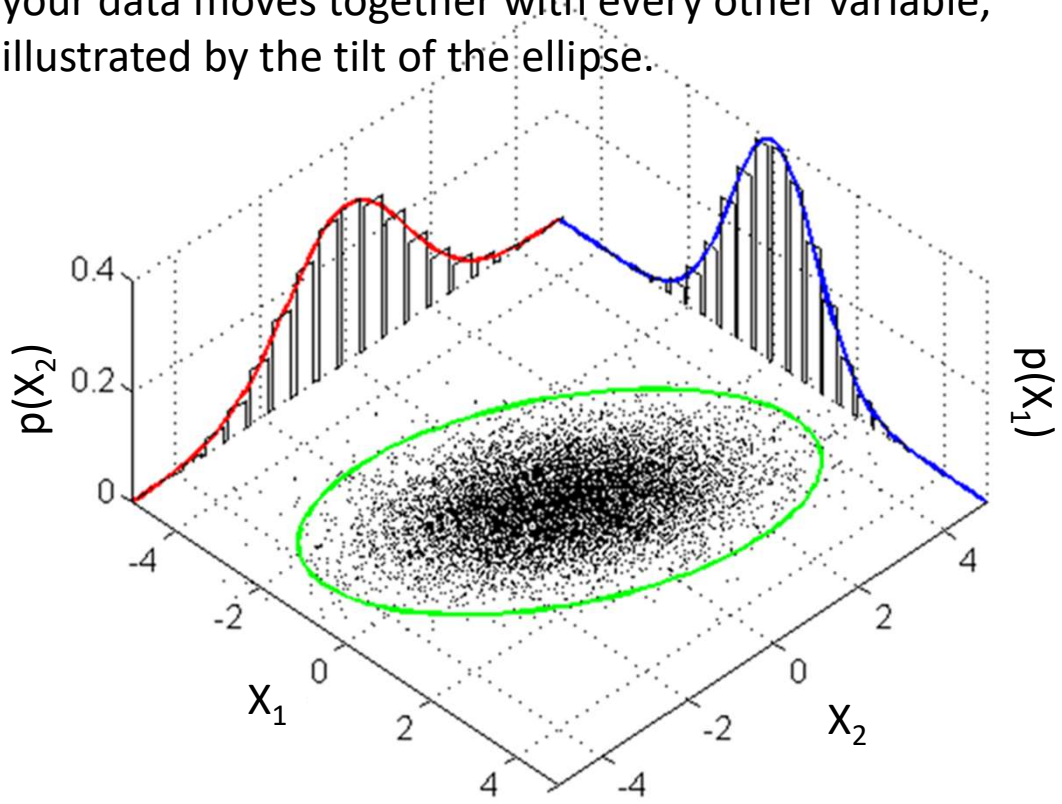
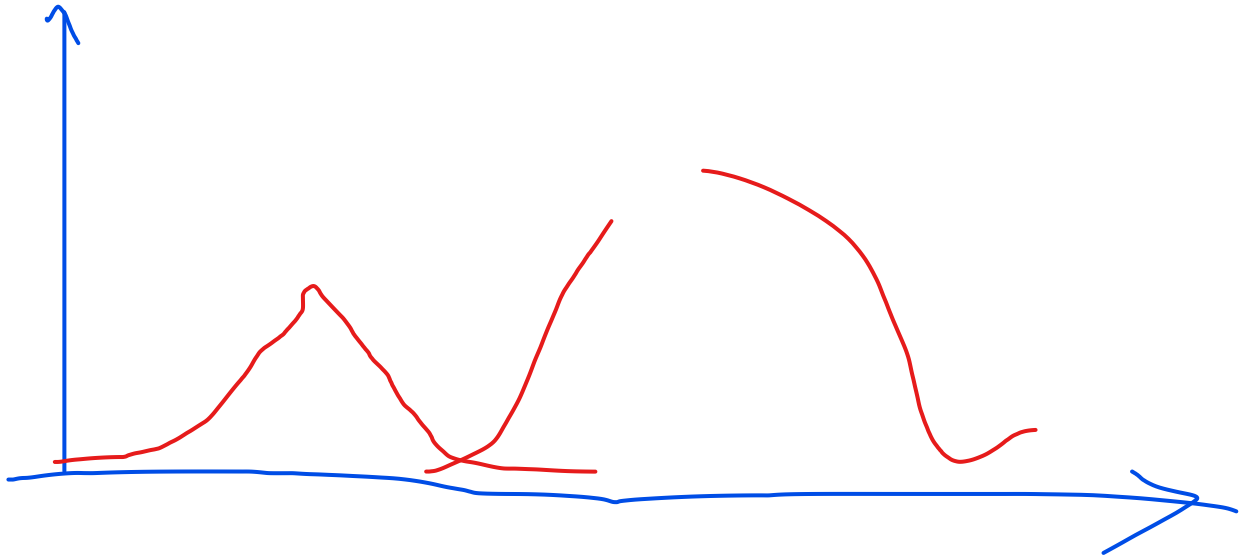


Figure source: https://en.wikipedia.org/wiki/Multivariate_normal_distribution

Linear Discriminant Analysis (LDA)

Approach: We model the distribution of X in each of the classes separately, and the use of Bayes Theorem to flip things around to obtain probability of y given x P(Y|X)

When we use normal (Gaussian) distribution for each class, this leads to linear or quadratic discriminant analysis. However, this approach is quite general, and other distribution can be used as well.
We will focus on normal distributions.



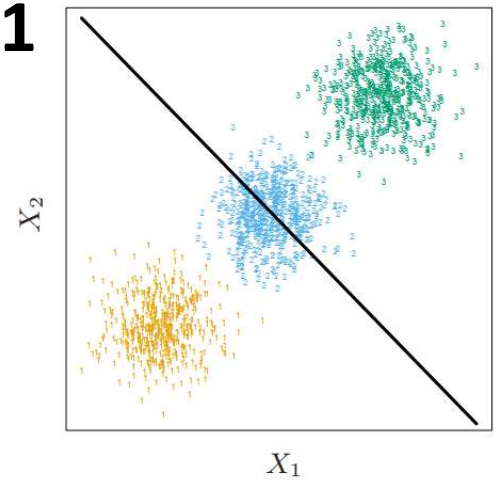
Bayes theorem to calculate the posterior probability:
Pr(Y= k|X =x) = (Pr(X=x|y=k).Pr(Y=k))/(Pr(X=x))

Written for LDA: Pr (Y =k| X=x) = f_k(X). pi(_k)/SUM(f_l(x).Pi(_e)
With pi_k= Pr (Pr

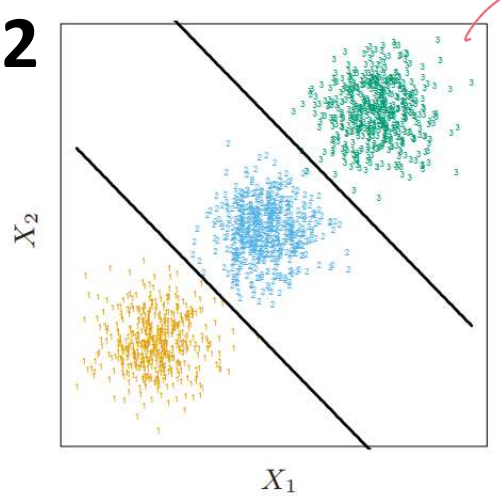
Quiz question

The following data points have been separated by decision boundaries. Which algorithm was used to produce the following outcome?

1



2



More than 2 classes → LR is not a good choice.

Multinomial logistic regression for 2 classes.

a) LDA for 1 and linear regression for 2

b) Linear regression for 1 and LDA for 2 → LDA: capture the distribution.

c) Multinomial logistic regression for 1 and linear regression for 2

d) Linear regression for 1 and multinomial logistic regression for 2

Figure: Introduction to Statistical Learning, T. Hastie et al. 2021

$$Pr(Y=k \mid X=x) = \frac{f_k(x) \cdot \pi_k}{\sum_{l=1}^{K} f_l(x) \cdot \pi_l}$$

Linear Discriminant Analysis with K=2 classes

Using the Gaussian density function with μ_k being the mean of class k & assuming that all σ_k are the same for all classes k

Then, $$f_k(x) = \frac{1}{\sqrt{2\pi}\,\sigma_k} e^{-\frac{1}{2}\left(\frac{x-\mu_k}{\sigma_k}\right)^2}$$

We want to find an equation for so-called discriminant score $\delta_k(x)$ (delta) to assign X to the class with the highest discriminant score.

The thinking behind the score for comparing classes k & l is following:

$$\log\left[\frac{Pr(Y=k \mid X=x)}{Pr(Y=l \mid X=x)}\right] = \log\frac{f_k(x)}{f_l(x)} + \log\frac{\pi_k}{\pi_l} \quad (*)$$

plugging $f_k(x)$ into the equation (*) performing simplification & cancellation we obtain delta

$$\delta_k(x) = x \cdot \frac{\mu_k}{\sigma^2} - \frac{\mu_k^2}{2\sigma^2} + \log(\pi_k)$$

Discriminant score.

Note that, $\delta_k(x)$ is a linear function of x. parameters (Gaussian distribution mean & sigma of class k?) → Through training data.

If $K = 2$ classes; $P_1 = P_2 = 0.5 \rightarrow$ one can see decision boundary is at

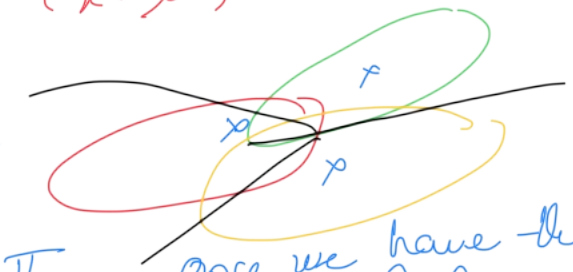
$$x = \frac{\mu_1 + \mu_2}{2}$$

Linear Discriminant Analysis with more than 2 classes

Gaussian density for more than 2 classes; with Σ being the covariance matrix.

$$f(x) = \frac{1}{(2\pi)^{p/2} |\Sigma|^{1/2}} e^{-\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu)}$$

Discriminant function again a linear function assuming the Σ are the same for each class.



Discriminant score $\delta_k(x) = x^T \Sigma^{-1} \mu_k - \frac{1}{2} \mu_k^T \Sigma^{-1} \mu_k + \log \pi_k$ (Σ^{-1} transverse). Once we have the $\delta_k(x)$ we can turn them into probabilities: $\hat{P}r(Y = k | X = x) = \frac{e^{\hat{\delta}_k(x)}}{\sum_{l=1}^{k} e^{\hat{\delta}_l(x)}}$

If $f_k(x)$ are Gaussian densities with the same covariance matrix in each class, the LDA can be used.

Quadratic Discriminant Analysis (QDA)

If $f_k(x)$ are Gaussian density with different covariance matrices in each class, QDA should be used.

$$\delta_k(x) = -\frac{1}{2}\log|\Sigma_k| - \frac{1}{2}(x-\mu_k)^T\Sigma_k^{-1}(x-\mu_k) + \log \pi_k$$

↗ Delta

If $f_k(x)$ being Gaussian densities with diagonal matrices (we expect complete independence features in real data)

⇒ Naive Bayes especially useful for large p (large features) where multivariate method like LDA & QDA breaks down.

Quiz question

The figure shows two non-linear decision boundaries to separate three classes.
What Algorithm could you use to do this?

a) LDA with X_1, X_2
b) QDA with X_1, X_2
c) Linear Regression with X_1, X_2, X_1X_2, X_1^2, X_2^2
d) LDA with X_1, X_2, X_1X_2, X_1^2, X_2^2

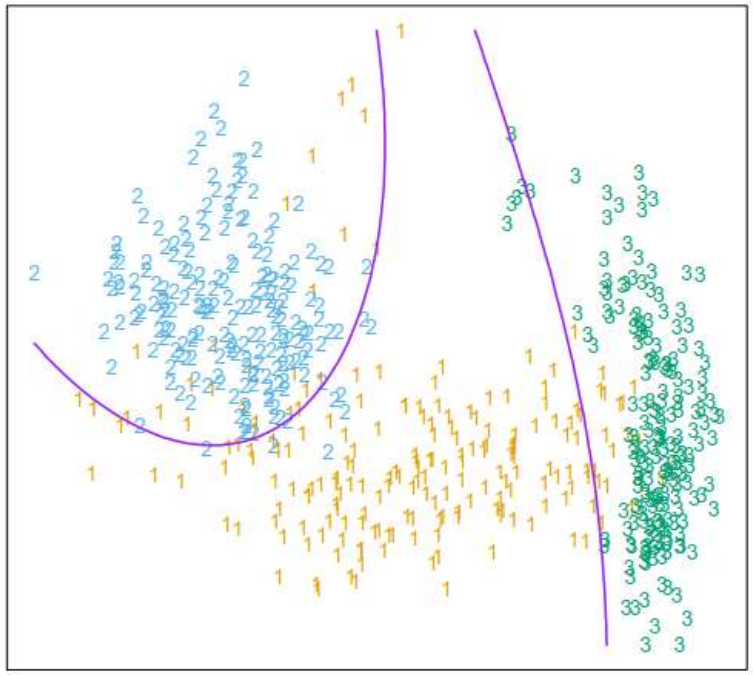


Figure: Introduction to Statistical Learning, T. Hastie et al. 2021

Quiz question

Decide which of the following statements are correct:

a) When the classes are well-separated, the parameter estimates for Logistic Regression are more stable than for LDA

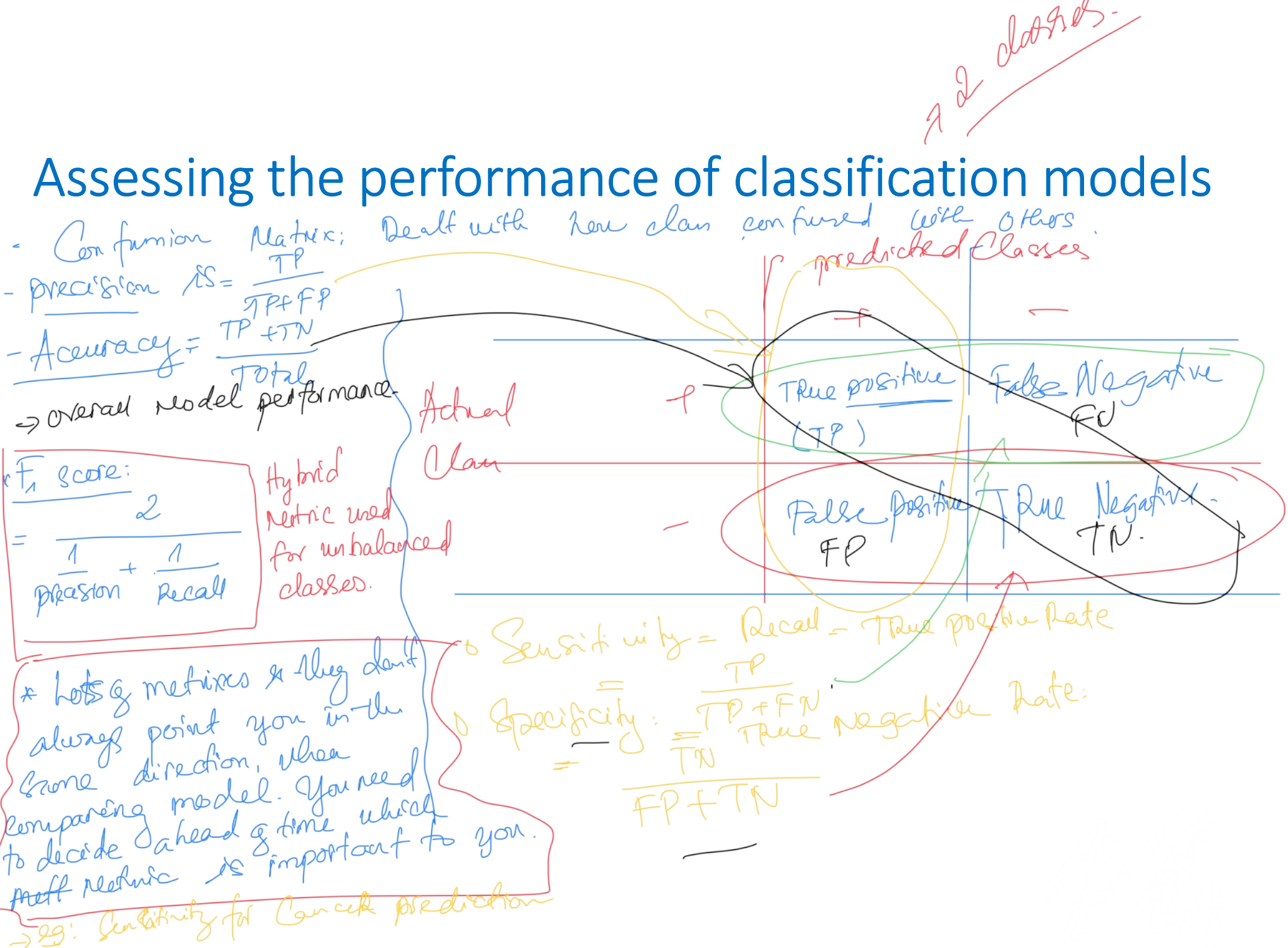
b) If n is small and the distribution of the predictors X is approximately normal in each of the classes, the LDA model is more stable than the logistic regression model.

c) LDA is popular when we have more than two response classes, because it also provides low-dimensional views of the data.

d) For non-linear decision boundaries KNN should be better than LDA or Logistic Regression as it is non-parametric but KNN doesn't tell which predictors are important

Performance Evaluation of Classification Algorithms

Assessing the performance of classification models



Quiz question

The following data table shows the prediction result of logistic regression on some test data. Create a confusion matrix from it in which positive refers to sickness. Calculate Accuracy, Sensitivity(Recall), Specificity, and Precision

Prediction	Actual status	Count
sick	sick	99
healthy	sick	1
healthy	healthy	8910
sick	healthy	990

Confusion Matrix

Actual \ Predicted	Positive	Negative
Positive	99	1
Negative	990	8910

Select the correct answer!

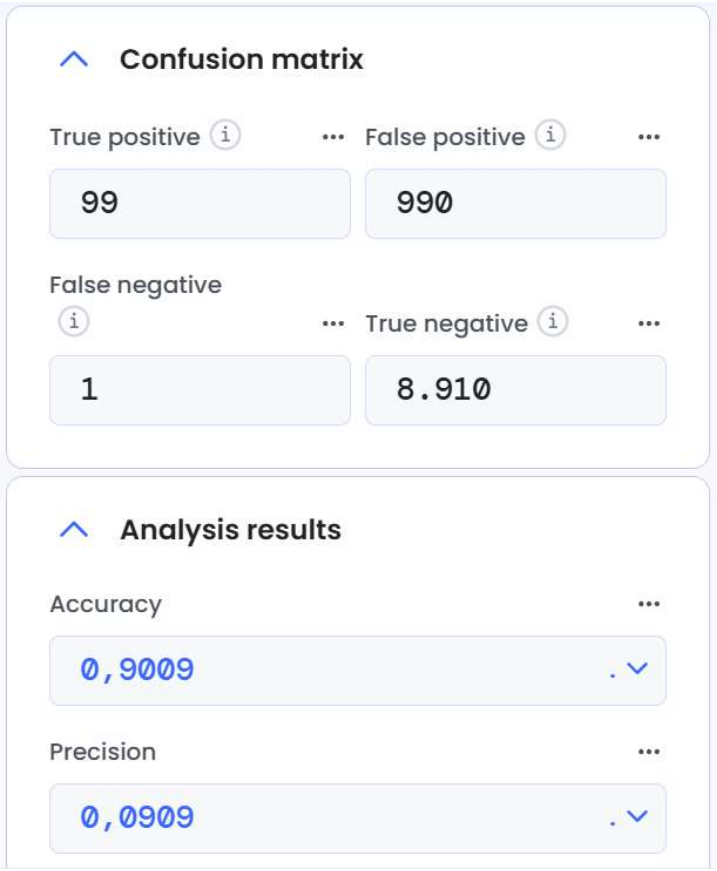
Answer	Accuracy	Recall	Specificity	Precision
a)	0.9009	0.9999	0.0909	0.9900
b)	0.9009	0.0909	0.9900	0.9999
c)	0.9009	0.9900	0.9000	0.0909
d)	0.9999	0.0909	0.9009	0.9900

0.9009 0.99 0.9

Confusion Matrix Calculators

Check the results which have calculated on the last slide!

https://www.omnicalculator.com/statistics/confusion-matrix



https://onlineconfusionmatrix.com/

	True Positive	True Negative
Predicted Positive	99	990
Predicted Negative	1	8910

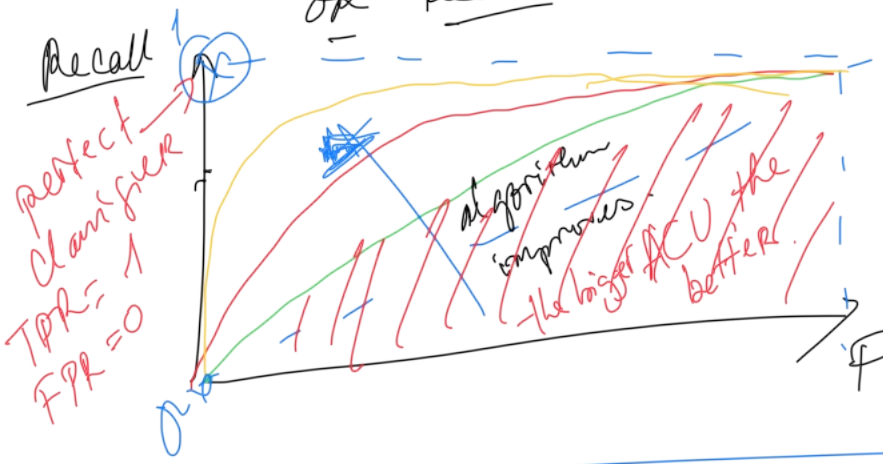
Measure	Value	Derivations
Sensitivity	0.9900	TPR = TP / (TP + FN)
Specificity	0.9000	SPC = TN / (FP + TN)
Precision	0.0909	PPV = TP / (TP + FP)
Negative Predictive Value	0.9999	NPV = TN / (TN + FN)
False Positive Rate	0.1000	FPR = FP / (FP + TN)
False Discovery Rate	0.9091	FDR = FP / (FP + TP)
False Negative Rate	0.0100	FNR = FN / (FN + TP)
Accuracy	0.9009	ACC = (TP + TN) / (P + N)
F1 Score	0.1665	F1 = 2TP / (2TP + FP + FN)
Matthews Correlation Coefficient	0.2843	TP*TN - FP*FN / sqrt((TP+FP)*(TP+FN)*(TN+FP)*(TN+FN))

The receiver operating characteristic (ROC)

ROC curve → a curve the performance of a binary classifier across all possible decision thresholds.

X_axis = False Positive Rate (FPR) = $\frac{\text{False Positive}}{\text{False positive + True Negative}}$.

Y_axis = True positive rate → sensitivity or Recall.

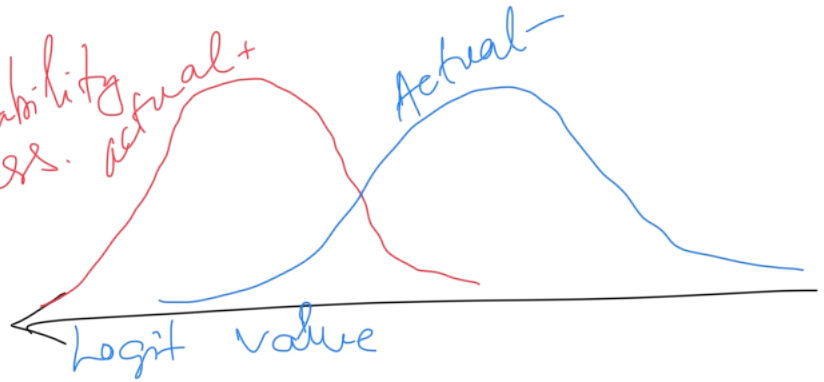


To generate an ROC curve for a given feature / algorithm, we first sort data table by ↓ logit values. We calculate the TPR & FPR in an accumulative fashion for each row in the data table. We plot the TPR over FPR.

Samples	logits	Class
4	0.98	1
15	0.97	1
⋮	0.94	1
		0
		⋮

→ TPR, FPR

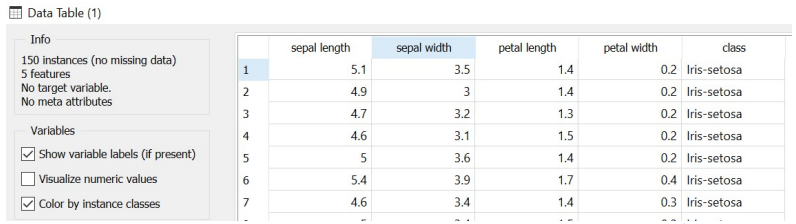
logit (P) = $\log\left(\frac{P}{1-P}\right)$ → predicted probability of positive class.

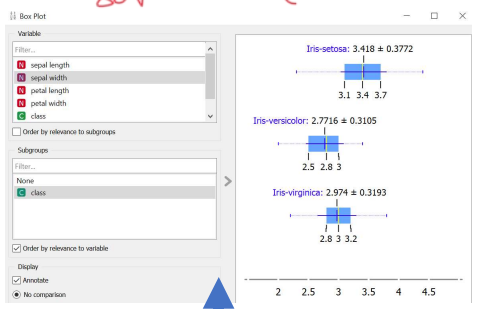


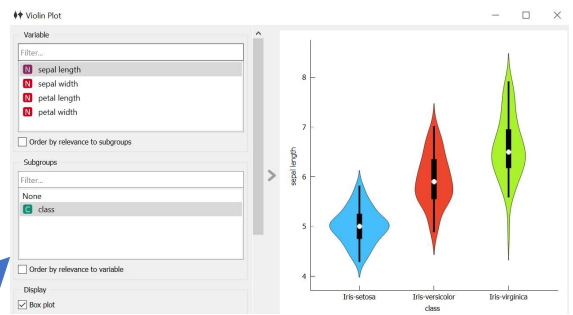
1. $\text{logit}(p) = \beta_0 + \beta_1 \cdot x_1 + \ldots + \beta_n x_n$ → logistic regression.
2. For NN, the last layer output (before activation) → logits, especially before. Softmax (Multiclass) or Sigmoid (Binary)

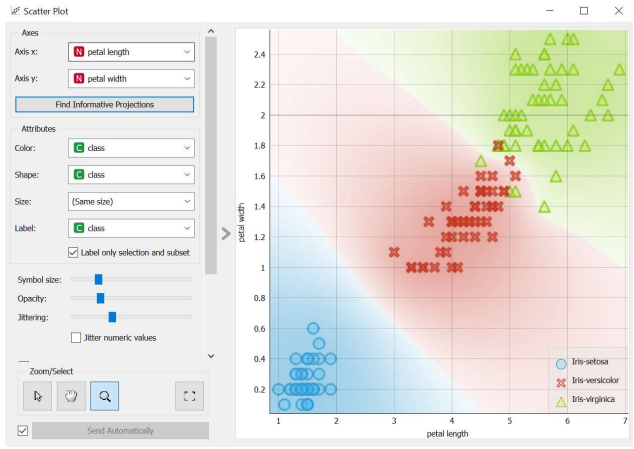
Working with Orange

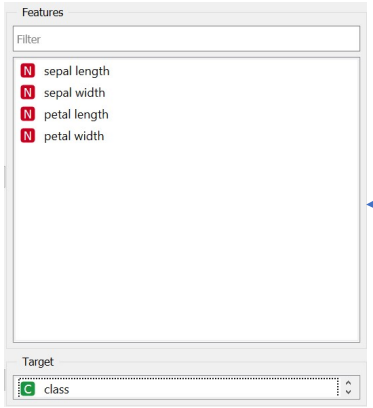
https://orangedatamining.com/download/

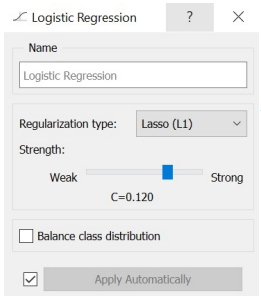






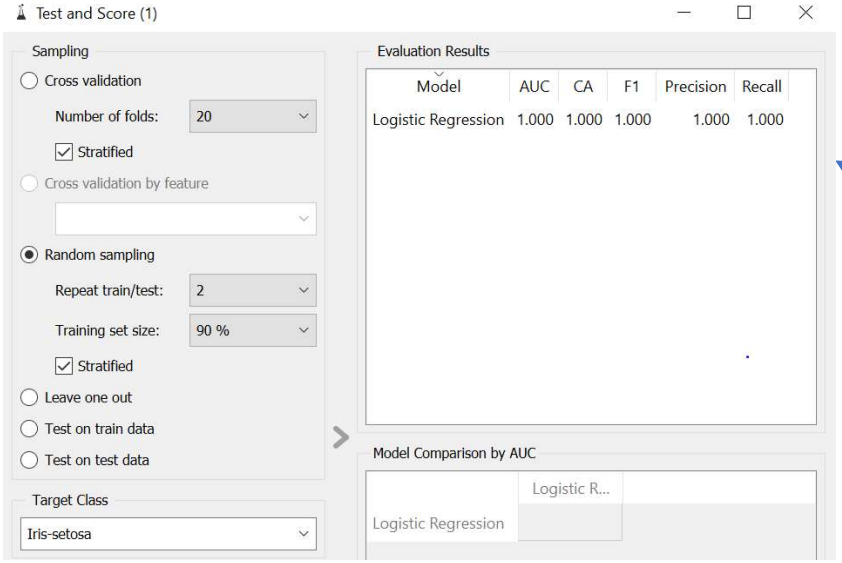


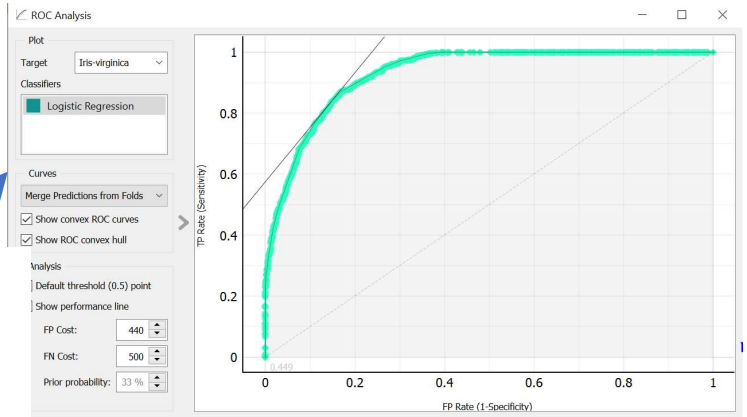


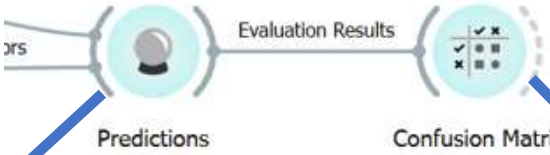


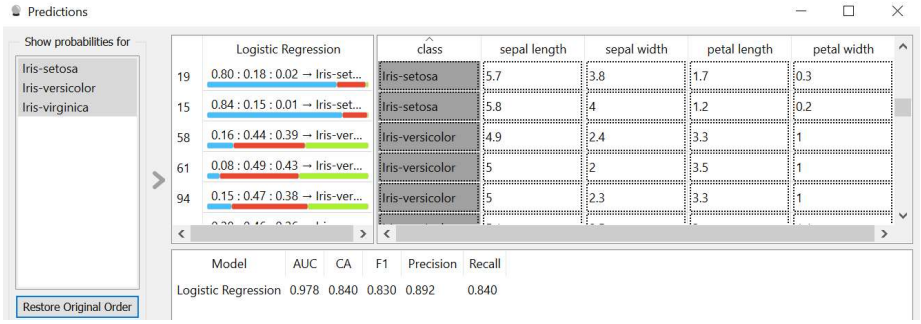
see next slide

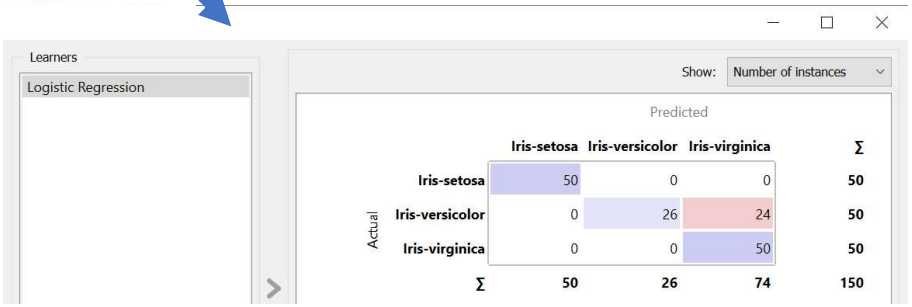
Working with Orange (continued)











Task: Please open Orange and reproduce the results on the previous two slides.