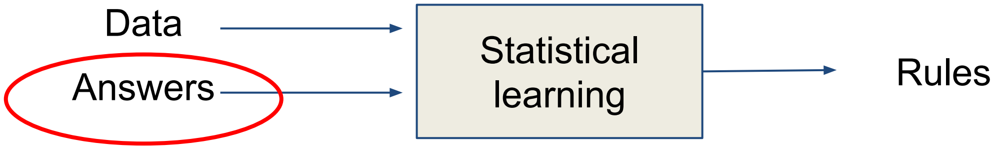
Supervised learning

Train the learners

Filippo Biscarini
Senior Scientist
CNR, Milan (Italy)

Nelson Nazzicari
Senior Scientist
CREA, Lodi (Italy)

Supervised learning problems



We know the answers (and feed them to the predictive algorithm)

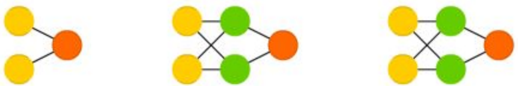
Why supervised?

Training examples

measured variables / features
on *n* examples

$$\begin{bmatrix} 1 \\ 0 \\ \vdots \\ 1 \end{bmatrix} = \begin{bmatrix} 0.12 & 1.5 & \dots & 0.9 \\ 2.05 & 0.95 & \dots & 1.1 \\ \vdots & \vdots & \ddots & \vdots \\ 3.5 & 0.88 & \dots & 1.75 \end{bmatrix}$$

labels / target variables (e.g. phenotypes) on n examples

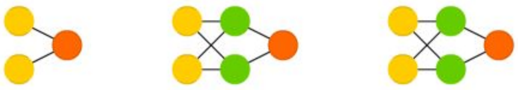


Unsupervised learning

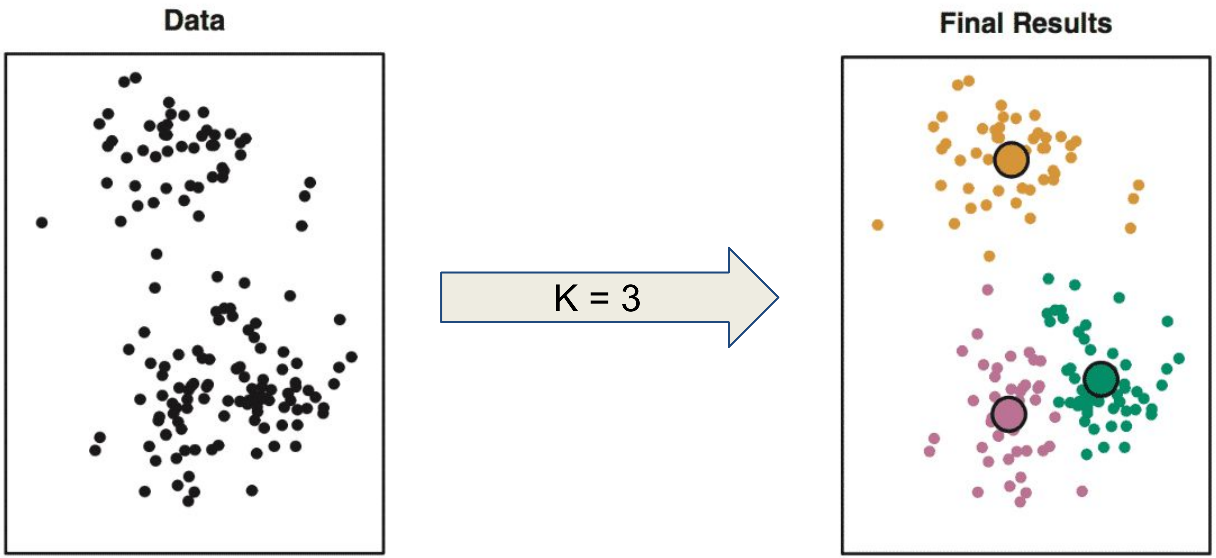
Training examples

measured variables / features
on n examples

$$\cancel{\begin{bmatrix} 1 \\ 0 \\ \vdots \\ 1 \end{bmatrix}} = \begin{bmatrix} 0.12 & 1.5 & \dots & 0.9 \\ 2.05 & 0.95 & \dots & 1.1 \\ \vdots & \vdots & \ddots & \vdots \\ 3.5 & 0.88 & \dots & 1.75 \end{bmatrix}$$



Unsupervised example: K-means clustering

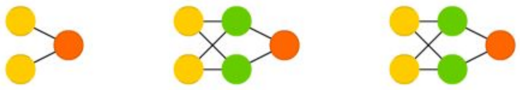


source:
https://www.iotforall.com/machine-learning-crash-course-unsupervised-learning

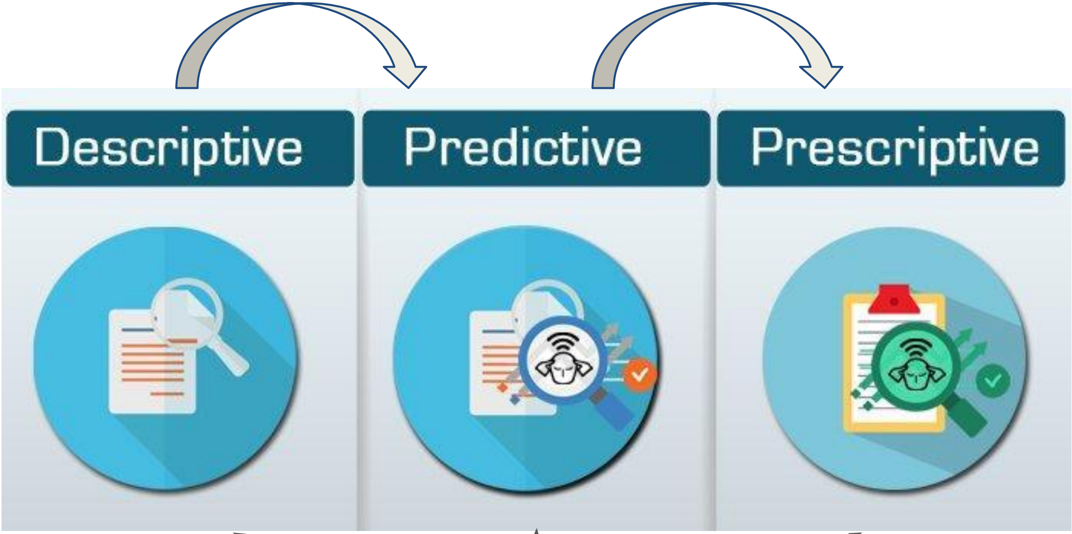
Supervised learning: Regression and classification

Supervised learning problems

- Regression (**predictive**) problems
- Classification (**predictive**) problems

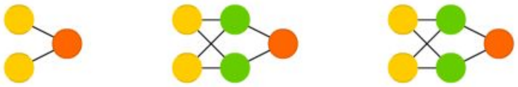


Supervised learning (predictive) problems



- Know the past
- Predict the future
- Act consequently

INFERENCE

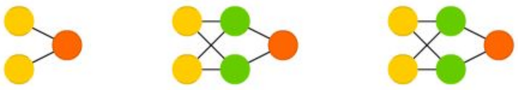


Supervised learning (predictive) problems

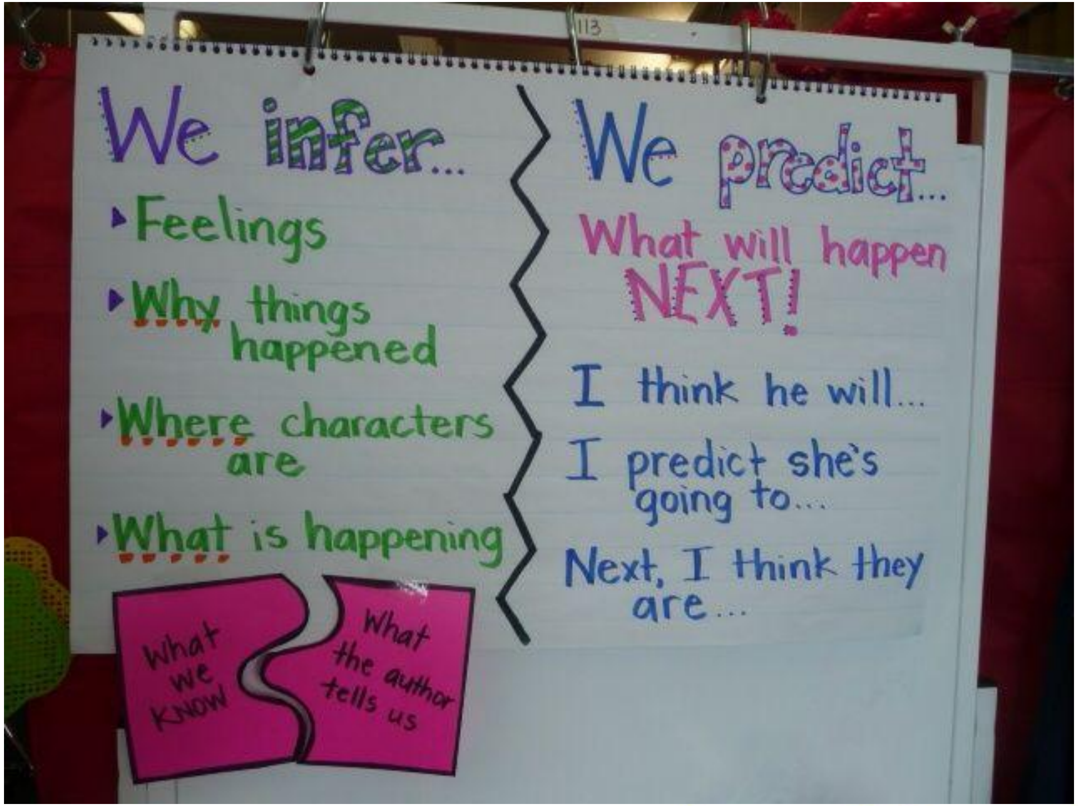
- Know the past
- Predict the future
- Act consequently

INFERENCE

- A catch-all term
- Can be confusing



Inference vs Prediction



- different statistical problems
- different objectives, different rules … different ballparks
- inference is in general more difficult than prediction

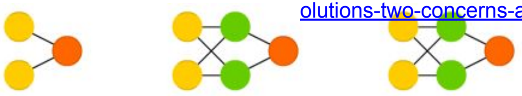
Supervised learning (predictive) problems

- Regression (**predictive**) problems
- Classification (**predictive**) problems

Predictive machines!

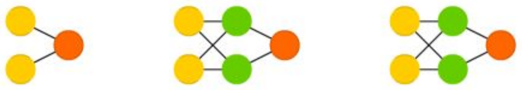
- Classifiers
- Regressors

source: https://blog.bigml.com/2013/03/12/machine-learning-from-streaming-data-two-problems-two-solutions-two-concerns-and-two-lessons/

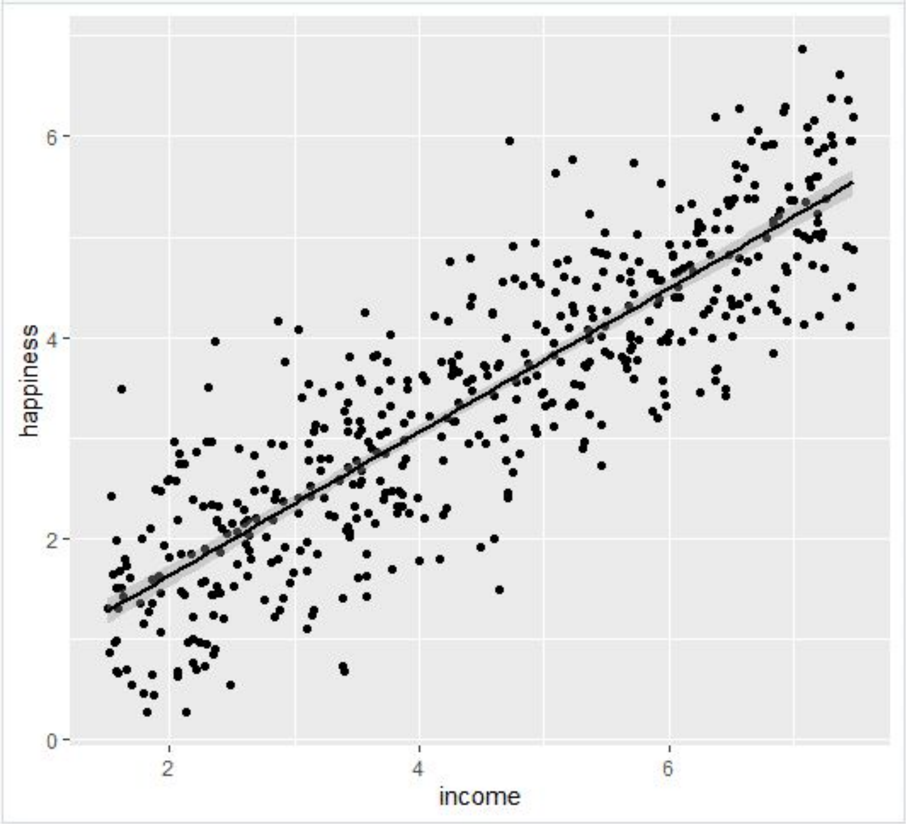


Regression problems

- the response variable **y** is **quantitative**
- e.g.: *height*, *weight*, *yield* (milk, crops), *blood sugar concentration*
- **y** = **target** (dependent) variable (a.k.a. response, objective variable)
- **X** = matrix of **features** (continuous, categorical)
- **regressor**: y = f(x) = **P(X)** ← [predictive machine]



Regression problems - simple regression



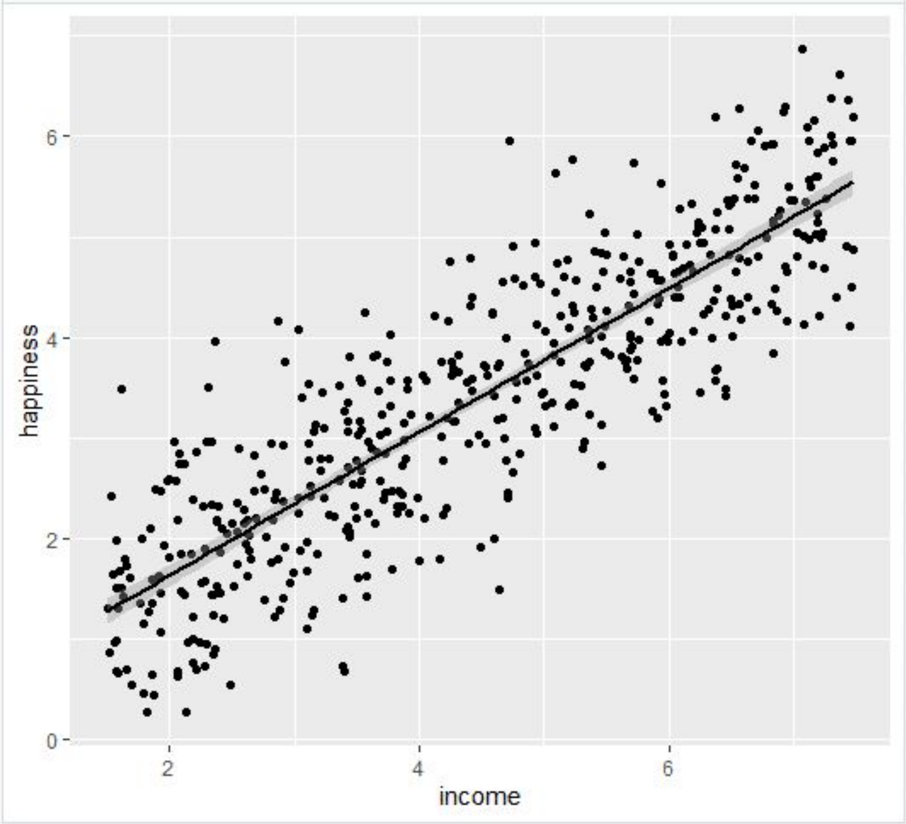
happiness = (intercept) + beta*income

or

income = (intercept) + beta*happiness

Source: https://www.scribbr.com/statistics/linear-regression-in-r/

Regression problems - simple regression



happiness = (intercept) + beta*income

or

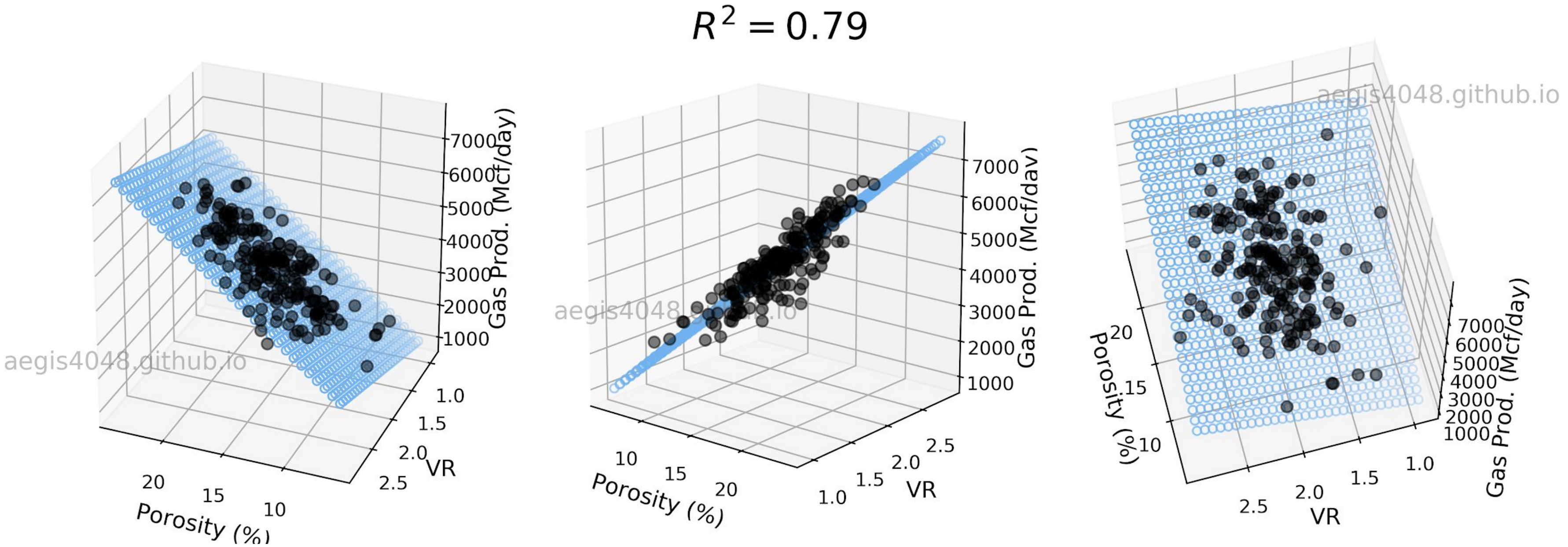
income = (intercept) + beta*happiness

cause → effect?

1. Can we predict backwards?
2. Can we make predictions if there is no cause-effect relationship?

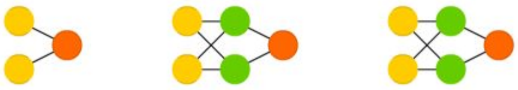
Source: https://www.scribbr.com/statistics/linear-regression-in-r/

Regression problems - multiple regression



- Gas production
- Well porosity %
- Vitrinite reflectance %

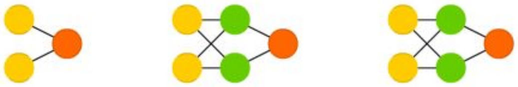
Source: https://aegis4048.github.io/mutiple_linear_regression_and_visualization_in_python



Multiple linear regression

$$y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p + \epsilon$$

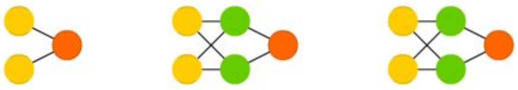
- y: target variable
- **β**’s: model coefficients
- X’s: features (predictors, independent variables, factors)



Multiple linear regression

$$\mathbf{y} = \beta\mathbf{X} + \mathbf{e}$$

- matrix (compact) notation
- vectors of observations (**y**), coefficients (**β**) and residuals (**e**)
- matrix of features (**X**)



Multiple linear regression

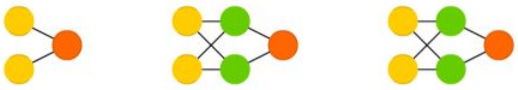
$$\mathbf{y} = \beta\mathbf{X} + \mathbf{e}$$

estimation of coefficients (learning!)

$$\hat{\mathbf{y}} = \hat{\beta}\mathbf{X}$$

→ **predictions!**

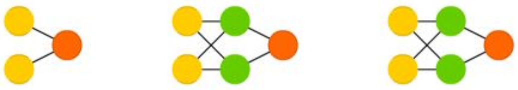
- matrix (compact) notation
- vectors of observations (**y**), coefficients (**β**) and residuals (**e**)
- matrix of features (**X**)



Predictions

$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2 + \ldots + \hat{\beta}_p x_p$$

with the estimated coefficients $\hat{\beta}$ and the feature values **X** we obtain the predicted values $\hat{y}$



Predictions

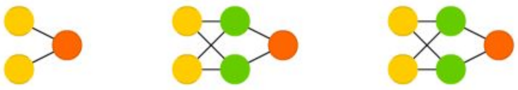
$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2 + \ldots + \hat{\beta}_p x_p$$

with the estimated coefficients $\hat{\beta}$ and the feature values **X** we obtain the predicted values $\hat{y}$

→ **how do we find the values for the the model coefficients β?**

(remember: the DL/ML models must learn the rules!)

(think about this: answer tomorrow)



Supervised learning recap

- **deep learning** is one of many methods that can be used to solve supervised learning problems
- deep learning is mainly used in **predictive problems**
- (can be used though also for inferential problems and for unsupervised learning)
- we'll see later how to use deep learning to solve linear regression