

Supervised learning

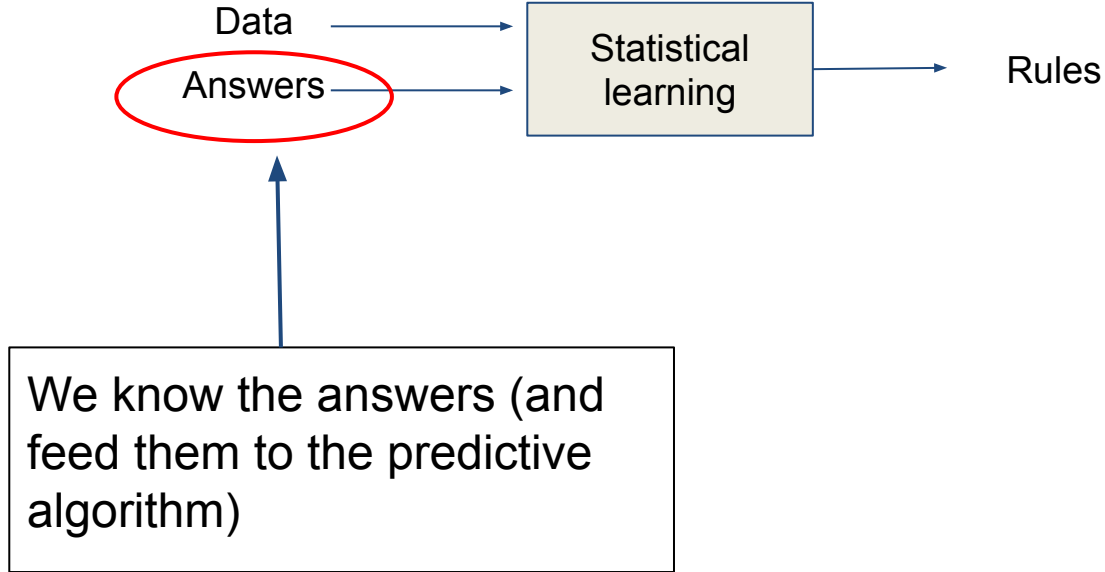
Train the learners

Filippo Biscarini
Senior Scientist
CNR, Milan (Italy)

Nelson Nazzicari
Research fellow
CREA, Lodi (Italy)



Supervised learning problems



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

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



Regression and classification

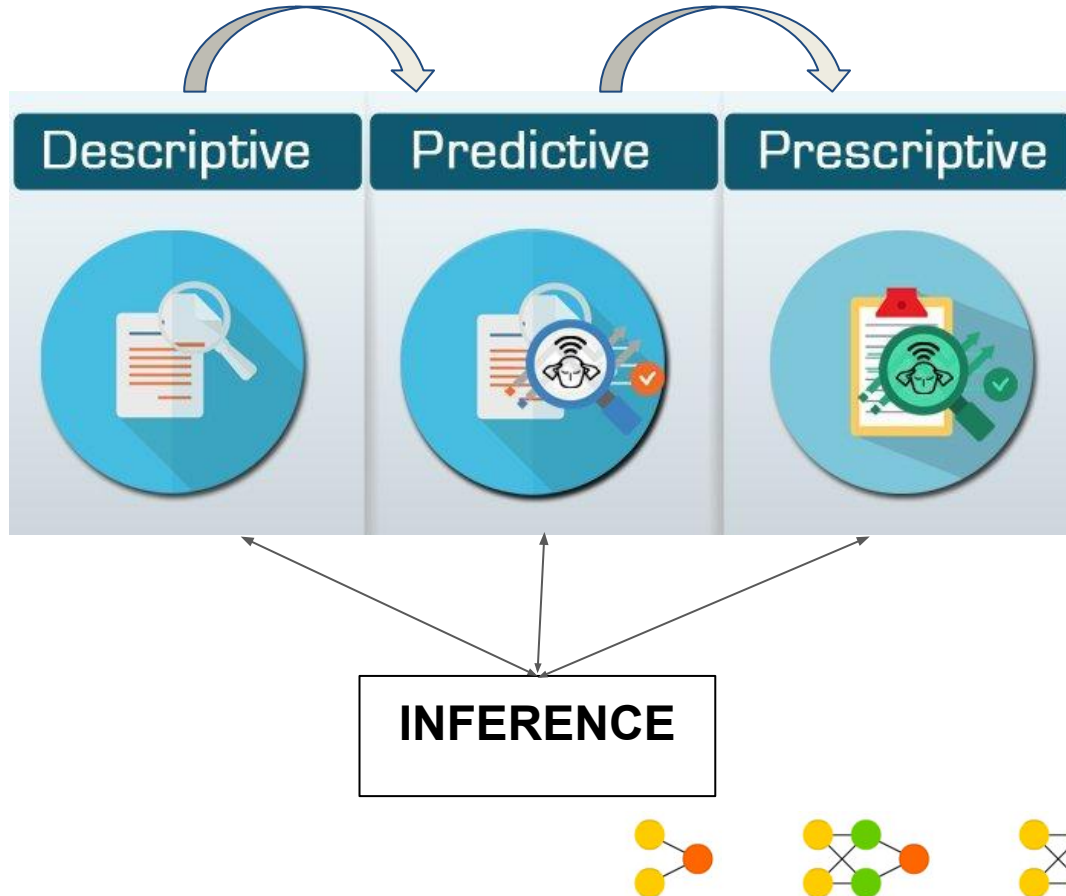


Supervised learning problems

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

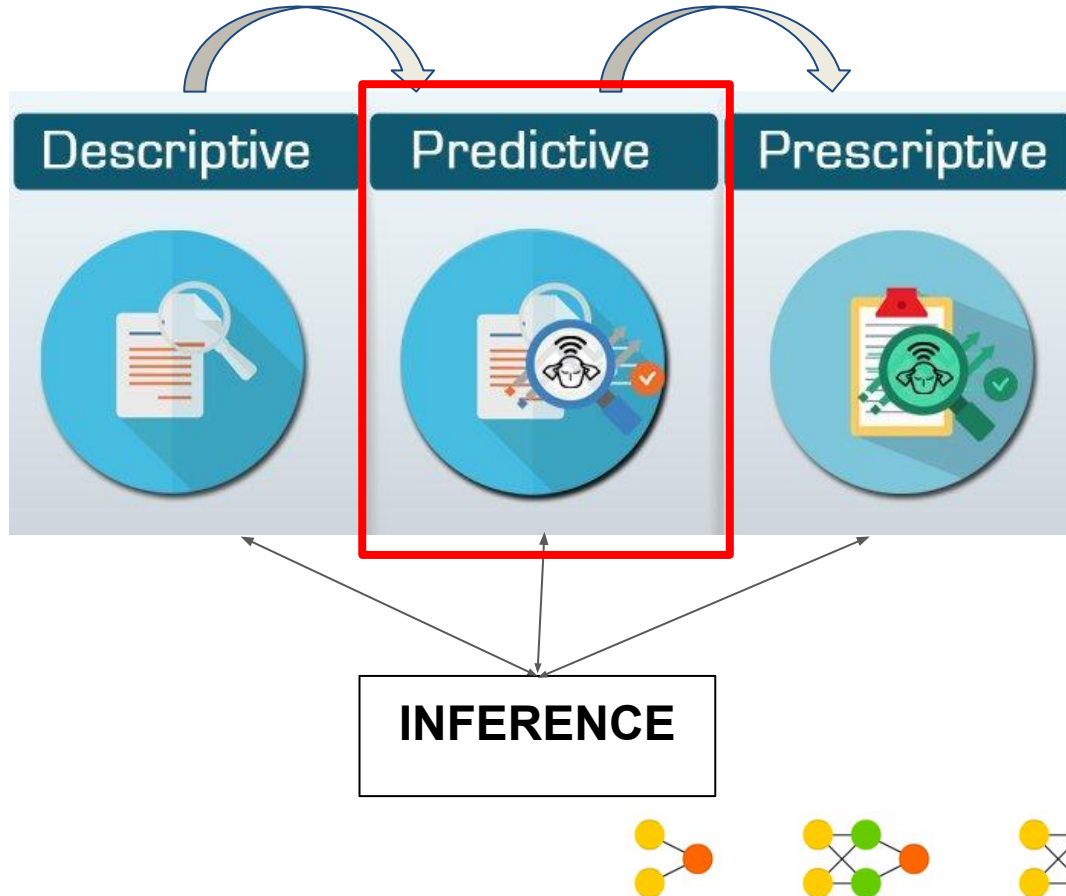


Supervised learning problems



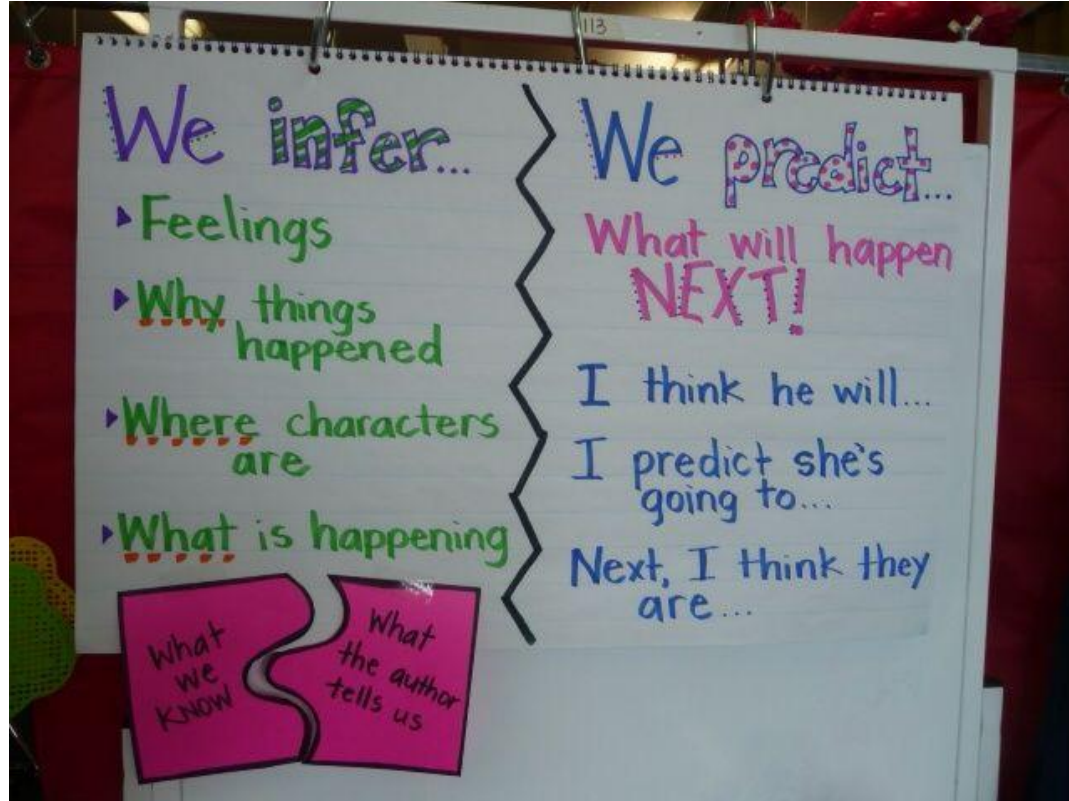
- Know the past
- Predict the future
- Act consequently

Supervised learning problems



- Know the past
- Predict the future
- Act consequently

Inference vs Prediction



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



Supervised learning problems

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

Predictive machines!

- Classifiers
- Predictors



source:

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

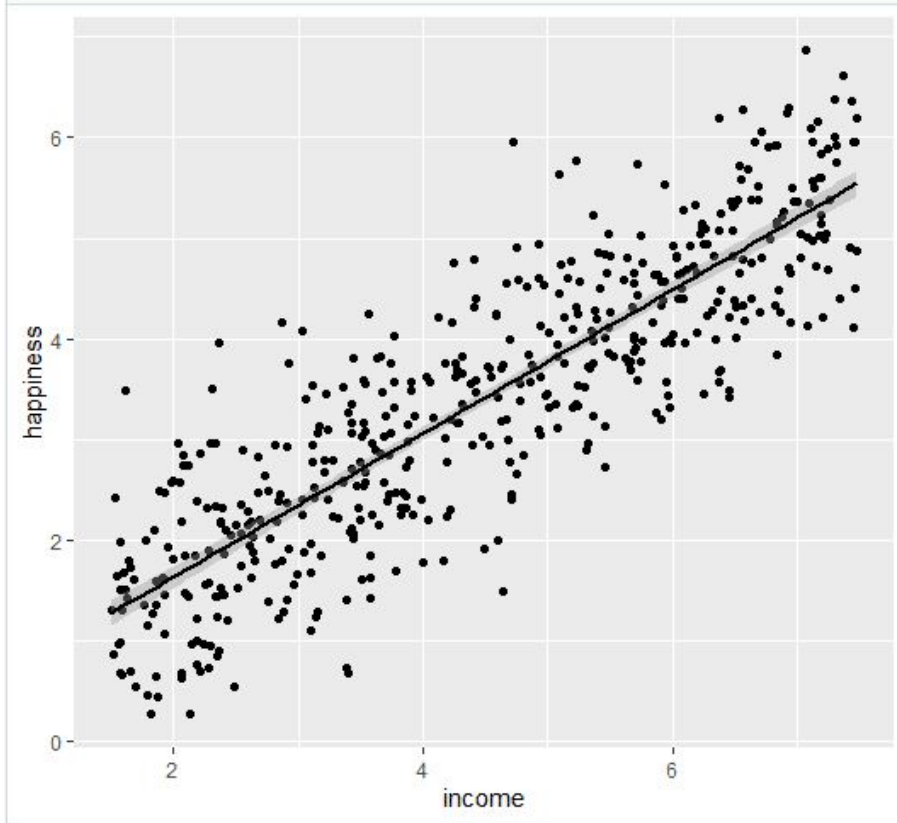
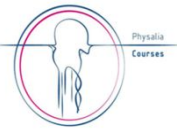


Regression problems

- 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)
- **predictor**: $y = f(x) = \mathbf{P}(\mathbf{X}) \leftarrow$ [predictive machine]



Regression problems - simple regression



$$\text{happiness} = (\text{intercept}) + \text{beta} * \text{income}$$

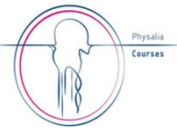
or

$$\text{income} = (\text{intercept}) + \text{beta} * \text{happiness}$$

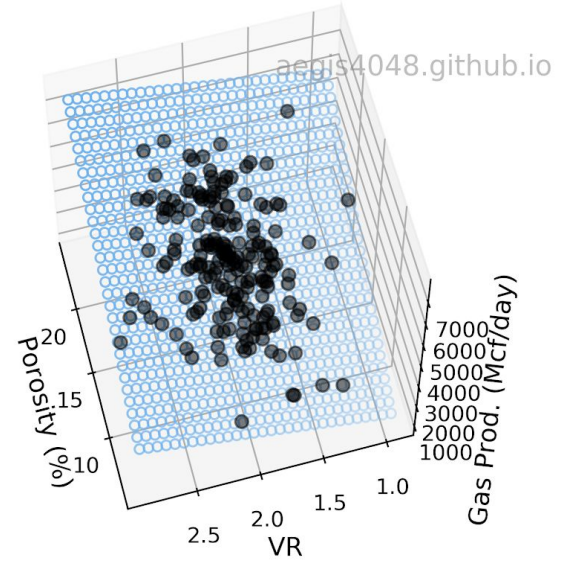
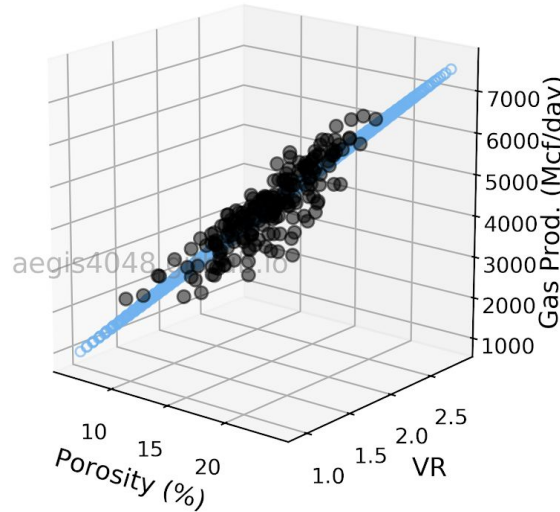
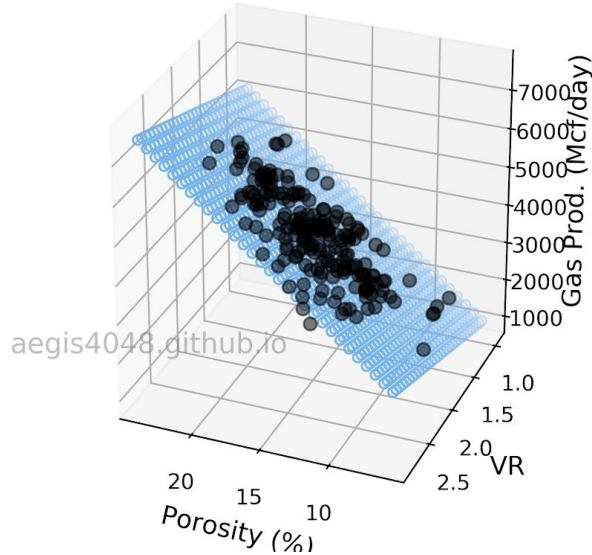
Source: <https://www.scribbr.com/statistics/linear-regression-in-r/>



Regression problems - multiple regression



$$R^2 = 0.79$$



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



Multiple linear regression

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

- 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 (\mathbf{y}), coefficients (β) and residuals (\mathbf{e})
- matrix of features (\mathbf{X})



Multiple linear regression

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

estimation of
coefficients

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

→ predictions!

- matrix (compact) notation
- vectors of observations (\mathbf{y}), coefficients (β) and residuals (\mathbf{e})
- matrix of features (\mathbf{X})



Predictions

$$\hat{y} = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p$$

with the estimated coefficients β and the feature values \mathbf{X} we obtain the predicted values \hat{y}



Predictions

$$\hat{y} = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p$$

with the estimated coefficients β and the feature values \mathbf{X} we obtain the predicted values \hat{y}

→ **how do we obtain the model coefficients β ?**



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**
- (could be used though also for inferential problems and for unsupervised learning)
- we'll see later how to use deep learning to solve linear regression

