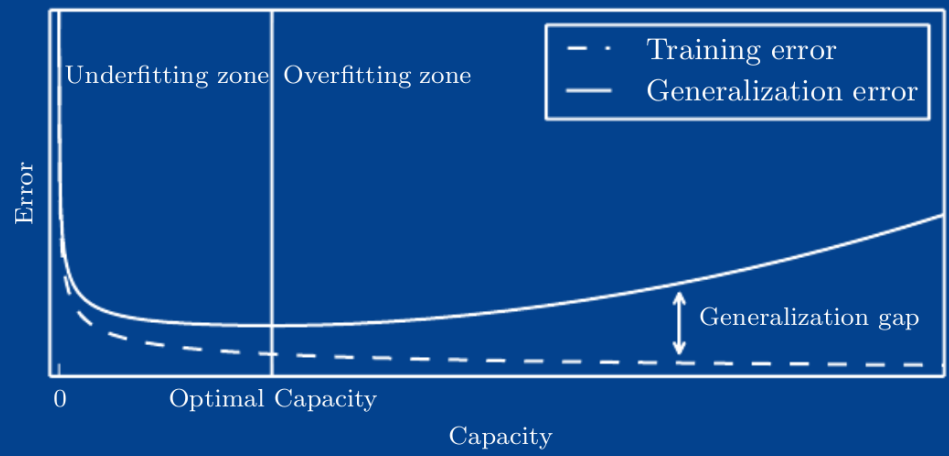
AARHUS
UNIVERSITY
SCHOOL OF ENGINEERING

# LESSON 7: Model-capacity, Under/over-fitting, Generalization

CARSTEN EIE FRIGAARD

SPRING 2021




"A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P , if its performance at tasks in T , as measured by P , improves with experience E." — Mitchell (1997).

# L07: Model-capacity, Under/over-fitting, Generalization

## Agenda

- Spørge-minutter..
- Resumé  af NN's.
- Demo af ML-systemer og klasse diskussion.
- Model Capacity,
- Under/over-fitting,
  Exercise: `L07/capacity_under_overfitting.ipynb`
- Generalization Error,
  Exercise: `L07/generalization_error.ipynb`

# RESUMÉ: GD

The numerically Gradient decent [GD] method is based on the gradient vector

$$\nabla_{\mathbf{w}} J(\mathbf{w})$$

for the gradient oprator

$$\nabla_{\mathbf{w}} = \left[ \frac{\partial}{\partial w_1}, \frac{\partial}{\partial w_2}, \ldots, \frac{\partial}{\partial w_m} \right]^{\top}$$

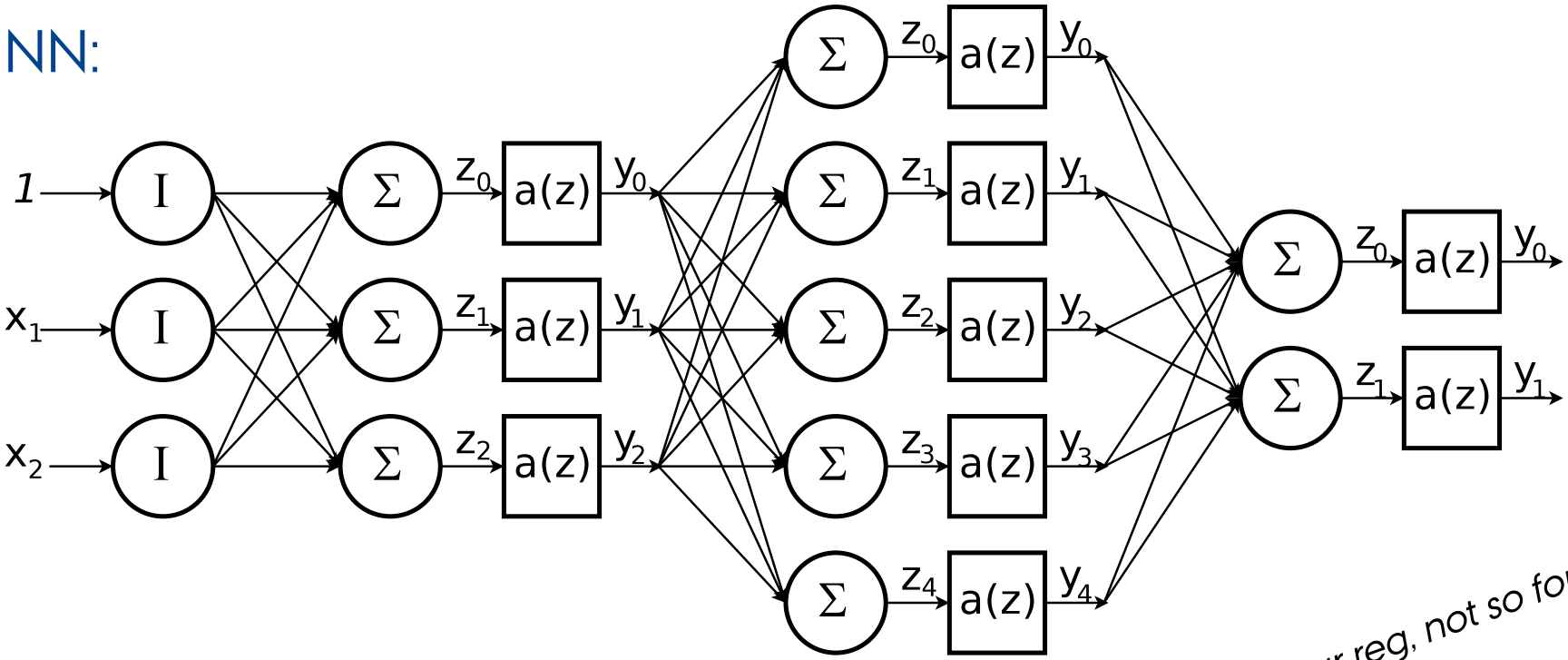
The algoritmn for updating via steps reads

$$\mathbf{w}^{(\text{next step})} = \mathbf{w} - \eta \nabla_{\mathbf{w}} J(\mathbf{w})$$

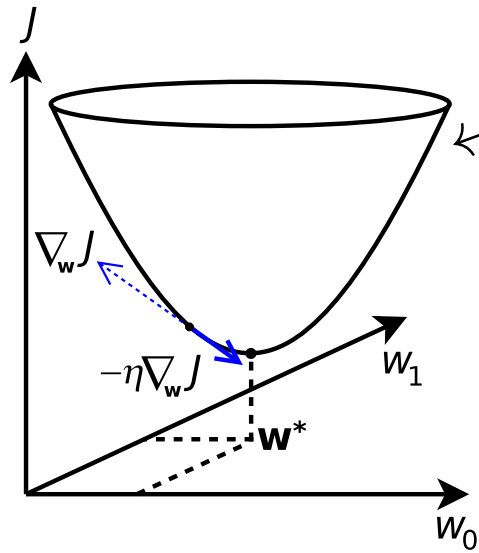
with $\eta$ being the step size.

# RESUMÉ: Training Deep Neural Networks

## NN:




## GD (via BPROP):




CONVEX ~ linear reg, not so for NNs

$$\mathbf{w}^{(\text{next})} = \mathbf{w} - \eta \nabla_{\mathbf{w}} J(\mathbf{w})$$

NOTE: NN: Neural net, GD: Gradient Descent, BPROP: Back Propagation

# Training Deep Neural Networks

*Equation 4-6. Gradient vector of the cost function*

$$\nabla_{\boldsymbol{\theta}} \mathrm{MSE}(\boldsymbol{\theta}) = \begin{pmatrix} \frac{\partial}{\partial \theta_0} \mathrm{MSE}(\boldsymbol{\theta}) \\ \frac{\partial}{\partial \theta_1} \mathrm{MSE}(\boldsymbol{\theta}) \\ \vdots \\ \frac{\partial}{\partial \theta_n} \mathrm{MSE}(\boldsymbol{\theta}) \end{pmatrix} = \frac{2}{m} \mathbf{X}^T (\mathbf{X}\boldsymbol{\theta} - \mathbf{y})$$

Notice that this formula involves calculation
set **X**, at each Gradient Descent step! This i
called *Batch Gradient Descent*: it uses the w
data at every step (actually, *Full Gradient D*
be a better name). As a result it is terribly sl
ing sets (but we will see much faster Gradient Descent algorithms
shortly). However, Gradient Descent scales we
features; training a Linear Regression model
dreds of thousands of features is much fa
Descent than using the Normal Equation or S

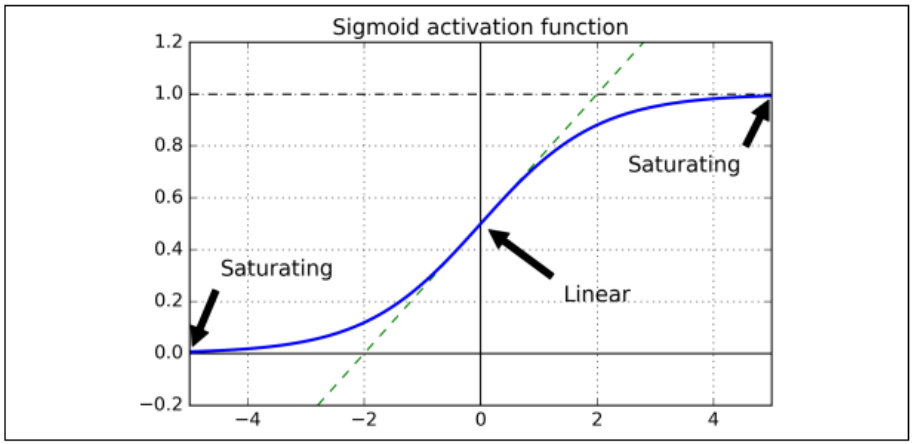



*Figure 11-1. Logistic activation function saturation*

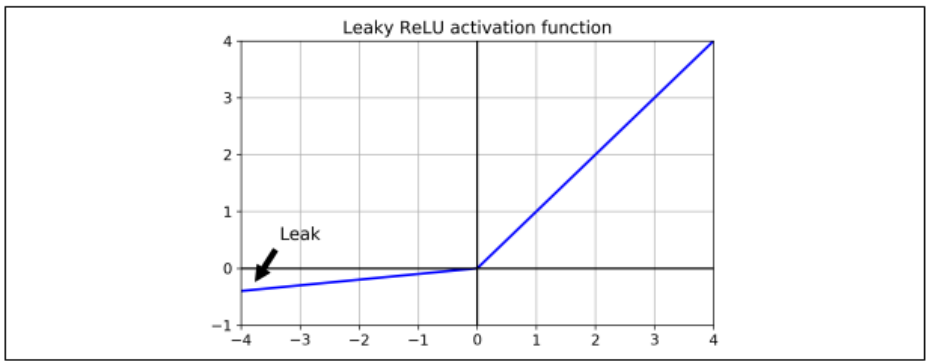



*Figure 11-2. Leaky ReLU*

Once you have the gradient vector, which points uphill, ju
tion to go downhill. This means subtracting $\nabla_{\boldsymbol{\theta}}\mathrm{MSE}(\boldsymbol{\theta})$
learning rate $\eta$ comes into play:[6] multiply the gradient v
size of the downhill step (Equation 4-7).

*Equation 4-7. Gradient Descent step*

$$\boldsymbol{\theta}^{(\text{next step})} = \boldsymbol{\theta} - \eta \nabla_{\boldsymbol{\theta}} \mathrm{MSE}(\boldsymbol{\theta})$$

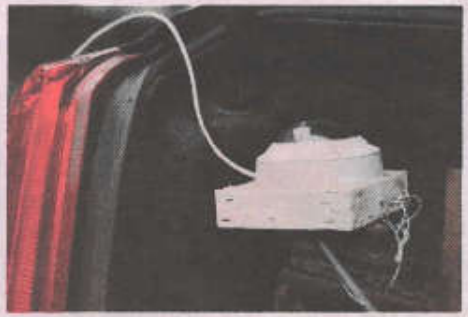
$$\mathbf{w}^{(\text{next})} = \mathbf{w} - \eta \nabla_{\mathbf{w}} J(\mathbf{w})$$


SmartBus
Powered by holo
Hop ombord
på Danmarks første
førerløse bus.










»Det er et forsøg på at skabe en ny brugerflade for at skrive. På en måde har jeg jo skrevet en roman med en bil,« fortæller Ross Goodwin om sin AI-forfattede bog *1 the road*

# På roadtrip med en insekthjerne

**Af MIKKEL BORIS**

I 2016 slog computeren AlphaGo den 18-dobbelte verdensmester i brætspillet Go, Lee Sedol. Go er et kompliceret og abstrakt spil, som kræver intuition og kreativitet, men den kunstige intelligens vandt med en række innovative træk overlegent.

Undervejs i spillet troede kommentatorerne

Goodwin til Weekendavisen fra sin lejlighed i Los Angeles.

Inden køreturen havde han brugt måneder på at træne maskinen. Han satte den til at læse et stort korpus af moderne litteratur fra hele verden, så den kunne lære at skrive af de store forfattere.

»Det fungerer ligesom autokorrekturen på din telefon, bare klogere og trænet på en mere litterær kilde. Den skriver bogstav for bogstav, så den har lært sig selv at forudsige det næste

når du dekonstruerer dem. Efter at have læst den i ét stræk og fået turen lidt på afstand har romanen fået en universalitet, så jeg kan projicere mine egne oplevelser ind i teksten,« uddyber Goodwin.

*– Du har beskrevet projektet som at lære en insekthjerne at skrive. Hvad betyder det?*

»Jeg forsøgte at pointere, at maskinen ikke er på niveau med den menneskelige hjerne. Et artificielt neuralt net er en algoritme, der er lavet, som vi tror, hjerner fungerer. Jeg synes

# Hvad siger pressen?

BLOG | Erik David John

## GPT-3 er ikke stærk AI

Erik David Johnson Tirsdag, 9. marts 2021 - 15:00 11

FØLG

Sidste år så vi GPT-3 melde sin ankomst på AI-scenen. For de af j
man med GPT-3 producere meningsfulde tekster i natursprog (r
uset niveau. Autogenererede sætninger og mindre artikler var s
menneskelig forfatter, og endnu en gang blev AI fremhævet som
vej til at ændre verden.

### Ikke så imponerende som det lyder

Sagen er bare at det ikke er så revolutionerende sc
skaberne af GPT-3 et øjeblik ville man forstå at der

### Om Erik David Jo

Erik er Principal AI Sp
arbejder med forretni

DR NYHEDER DRTV DR LYD

TEKNOLOGI

## Robotter kan nu lugte og mærke (næsten) ligesom mennesker

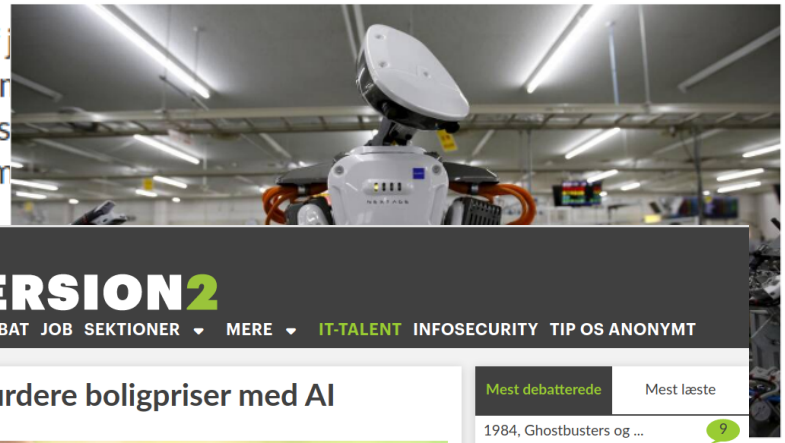
Sanserne er nødvendige, når robotter skal begå sig i menneskets kaotiske verden.

AF ESBEN HARDENBERG
26. JUN 2019 BEMÆRK: ARTIKLEN ER MERE END 30 DAGE GAMMEL

[?] FORKLAR ORD [AA] STØRRE TEKST [O] LÆS OP

I Odense er firmaet AmiNIC ved at udvikle en ny robot, med en ganske særlig evne: lugtesans. Robottten skal bruges til at lugte til fisk og afgøre, om fisken er frisk eller ej. Det kan hjælpe restauranter og fødevareindustrien til kun at smide de råvarer ud, som er fordærvede.

På den anden side af Atlanten, i Cambridge, på MIT-universitets afdeling for IT og



ING/VERSION2

NYHEDER BLOGS DEBAT JOB SEKTIONER MERE IT-TALENT INFOSECURITY TIP OS ANONYMT

## Banker vil vurdere boligpriser med AI

(Illustration: Puttachat Kumkrong // Bigstock)

Mest debatterede | Mest læste

1984, Ghostbusters og ... 9

Regningen for Smittestop-appen stiger 8

Frankrig og Holland støtter EU i kamp mod amerikanske it-giganter 2

Forslag om øget it-kontrol i det offentlige: »Det her stykke arbejde SKAL laves« 0

WEBINARS AND WHITEPAPERS

WEBINAR

Øget effektivitet, fleksibilitet og muligheder for


e Count: 8
e Count: 0
A T
[0 0]
0 0]
[0 0]
side
front
side
front
front

# A computer vision system to monitor the infestation level of Varroa destructor in a honeybee colony

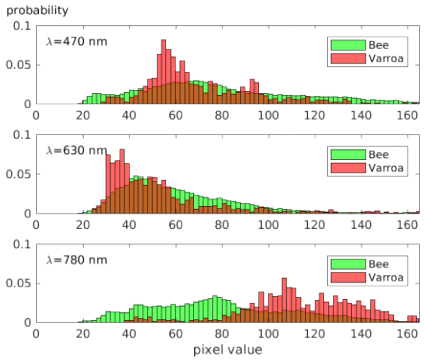



Figure 5: Histograms of bee and varroa pixel intensity values, for the spectral wavelengths 470 nm, 630 nm, and 780 nm respectively, recorded with the JAI camera. The image path is via the mirror-window-mirror, i.e. data were sampled with the setup given in figure 1. The image data for the histogram is the single bee with mite seen in figure 6.

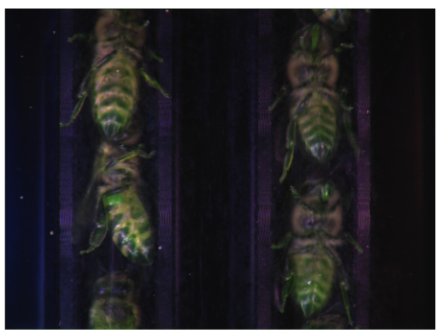


Figure 6: The actual unprocessed camera view of the bees

spectively. The CM analysis was able to rank all wavelengths combinations, using one, two, three or four district wavelengths to give a ranking list of 'best' combination also taking the JAI camera spectrum into account.

The CM value of the actual choose wavelengths combination (470-630-780 nm) gave a rank just below the CM average score. This CM analysis was conducted after picking the actual used wavelengths, so later versions of the VMU might want to investigate a CM combination with a higher rank.

A specially designed diffuser and a number of narrow spectral LE were mounted in the camera focal diffuse illuminat flections.

Figure 6 disp along the passa era, with the gre with the NIR m

*2.3.3. Real-time processin*

A color and tion of 1296×96 from the camer over two separat sary sustained b ing frames real-t

These data w line post-proces first matching t rally coalescing producing a 24 the later image

Lossless real-t can be applied bandwidth than

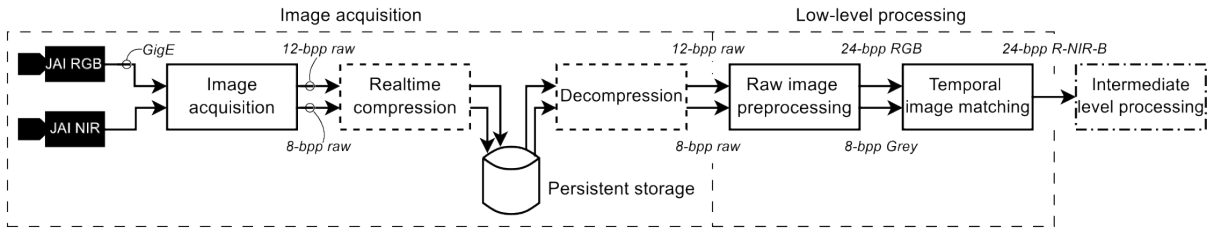



Figure 7: The low-level image processing pipeline. Raw camera images are stored on disk for later retrieval and post-processing. 12- and 8 bits-per-pixel are used as the raw JAI/Bayer packed pixel format for the RGB and IR images respectively. Lossless, real-time compression can be introduced if persistent storage bandwidth is less than the raw-stream image rate of 93 MiB/sec. The 12- and 8-bpp raw images from the network arrives out-of-order with respect to each other, hence the need for the temporal image matching.

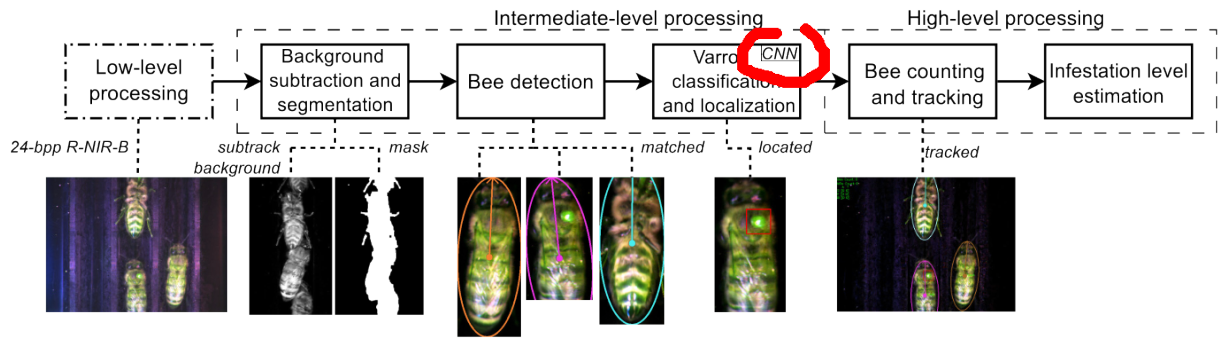
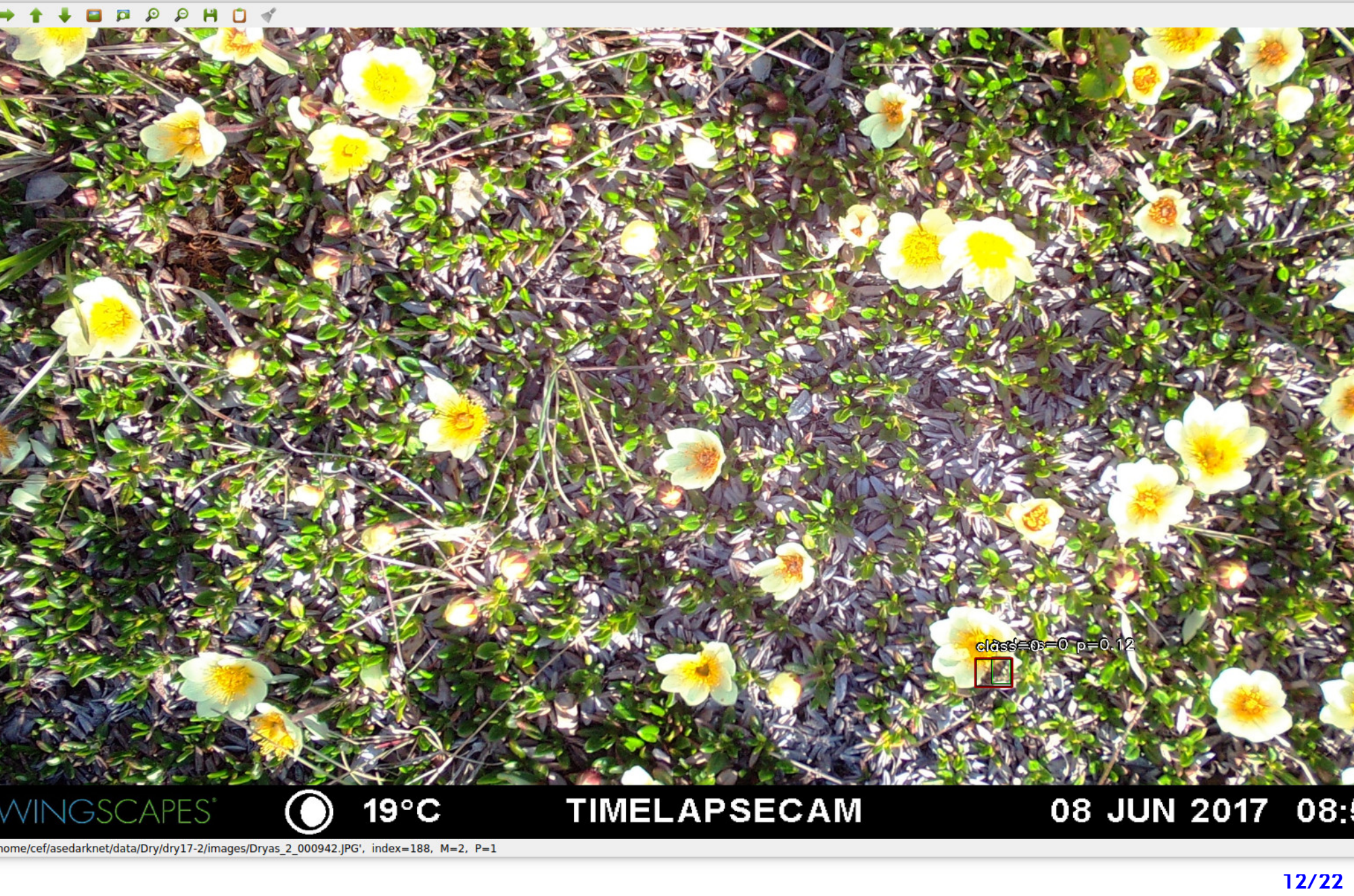



Figure 8: The processing pipeline of the intermediate- to high-level image processing algorithms to analyze and count the number of bees with *Varroa destructor*. A trained convolutional neural network (CNN) was used for the Varroa classification and localization stage.

# BA Project: generic annotation tool



# MODEL CAPACITY

# The Map

*ML algo+ model selection*

# RESUMÉ: L02/performance_metrics.ipynb

## Classification metrics

See the Classification metrics section of the user guide for further details.

| | |
|---|---|
| `metrics.accuracy_score` (y_true, y_pred[, ...]) | Accuracy classification score. |
| `metrics.auc` (x, y[, reorder]) | Compute Area Under the Curve (AUC) using the trapezoidal rule |
| `metrics.average_precision_score` (y_true, y_score) | Compute average precision (AP) from prediction scores |
| `metrics.cohen_kappa_score` (y1, y2[, labels, ...]) | Cohen's kappa: a statistic that measures inter-annotator agreement. |
| `metrics.confusion_matrix` (y_true, y_pred[, ...]) | Compute confusion matrix to evaluate the accuracy of a classification |
| `metrics.f1_score` (y_true, y_pred[, labels, ...]) | Compute the F1 score, also known as balanced F-score or F-measure |
| `metrics.log_loss` (y_true, y_pred[, eps, ...]) | Log loss, aka logistic loss or cross-entropy loss. |
| `metrics.precision_score` (y_true, y_pred[, ...]) | Compute the precision |
| `metrics.recall_score` (y_true, y_pred[, ...]) | Compute the recall |
| `metrics.roc_auc_score` (y_true, y_score[, ...]) | Compute Area Under the Receiver Operating Characteristic Curve (ROC AUC) from prediction scores. |
| `metrics.roc_curve` (y_true, y_score[, ...]) | Compute Receiver operating characteristic (ROC) |
| `metrics.zero_one_loss` (y_true, y_pred[, ...]) | Zero-one classification loss. |

## Regression metrics

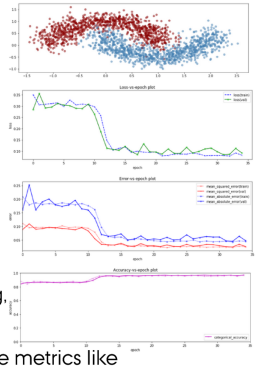
See the Regression metrics section of the user guide for further details.

| | |
|---|---|
| `metrics.explained_variance_score` (y_true, y_pred) | Explained variance regression score function |
| `metrics.max_error` (y_true, y_pred) | max_error metric calculates the maximum residual error. |
| `metrics.mean_absolute_error` (y_true, y_pred) | Mean absolute error regression loss |
| `metrics.mean_squared_error` (y_true, y_pred[, ...]) | Mean squared error regression loss |
| `metrics.mean_squared_log_error` (y_true, y_pred) | Mean squared logarithmic error regression loss |
| `metrics.median_absolute_error` (y_true, y_pred) | Median absolute error regression loss |
| `metrics.r2_score` (y_true, y_pred[, ...]) | R^2 (coefficient of determination) regression |

## Notes on Keras MLPs

Typical Keras MLP Supervised Classifier setup..

- loss function
  `loss='categorical_crossentropy'`
- metrics collected via history
  ```
  metrics=[
   'categorical_accuracy',
   'mean_squared_error',
   'mean_absolute_error'])
  ```
- input lay.: categorical encoding,
- output lay.: softmax function,

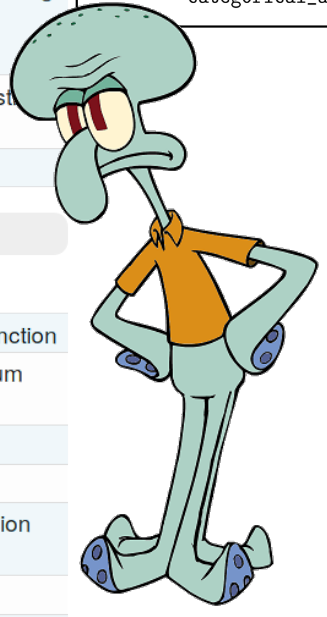


And notice that Keras do *not* provide metrics like

`precision, recall, F1`

but instead

`categorical_accuracy, binary_accuracy`



# Model capacity

Exercise: `capacity_under_overfitting.ipynb`

Dummy and Paradox classifier:
*capacity fixed* $\sim 0$, cannot generalize at all!

Linear regression for a polynomial model:
*capacity* $\sim$ degree of the polynomial, $x^n$

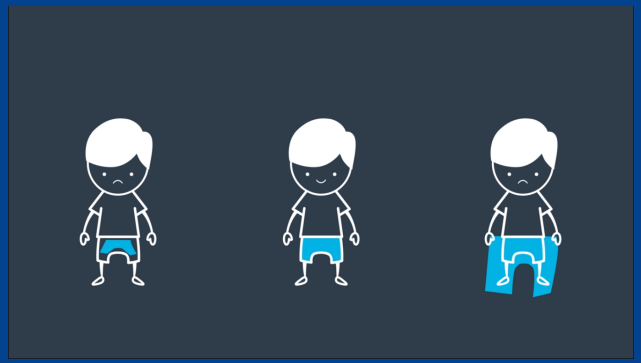
Neural Network model:
*capacity* $\propto$ number of neurons/layers

Homo sabiens ("modern humans"):
*capacity* $\propto$ the IQ 'score' function?

$\Rightarrow$ **Capacity** can be hard to express as a quantity for some models, but you need to choose..

$\Longrightarrow$ how to choose the **optimal** *capacity*?

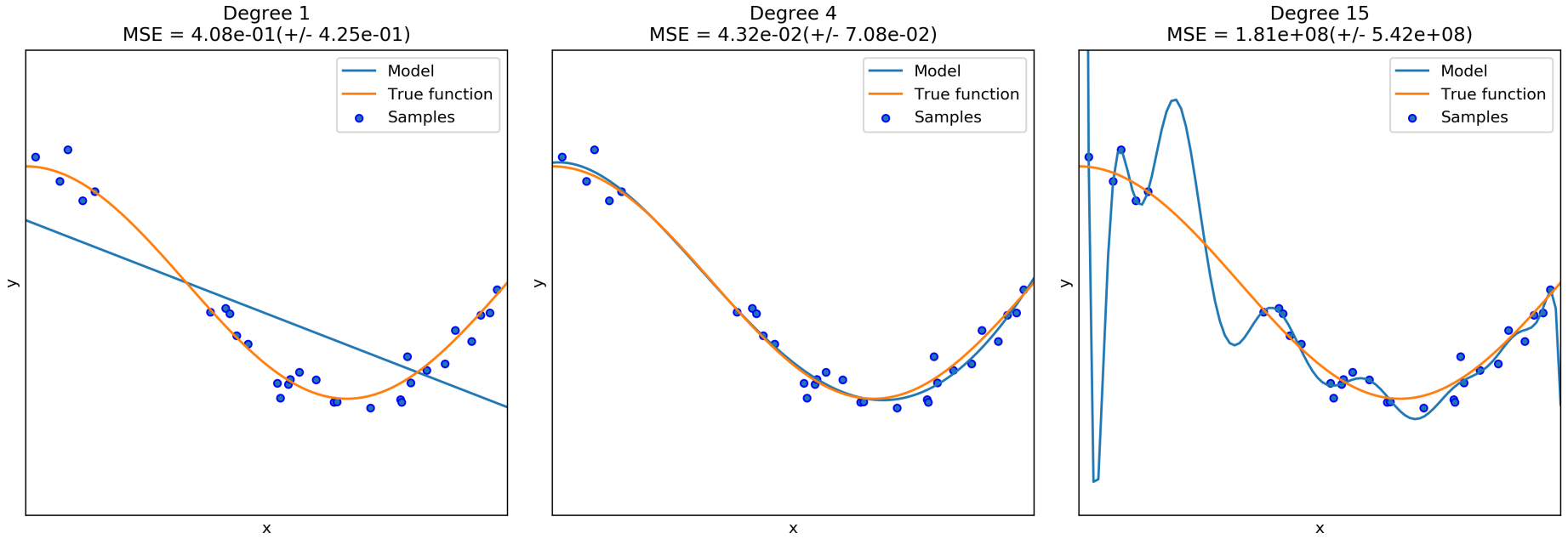
# UNDER- AND OVERFITTING



# Under- and overfitting

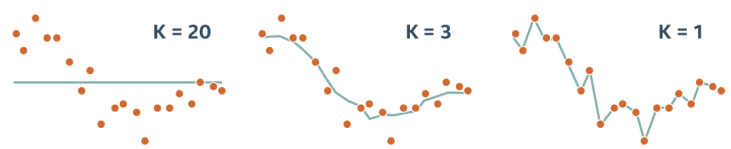
Exercise: `capacity_under_overfitting.ipynb`

Polynomial linear reg. fit for underlying model: `cos(x)`




- underfitting:
  capacity of model too low,
- overfitting:
  capacity to high.

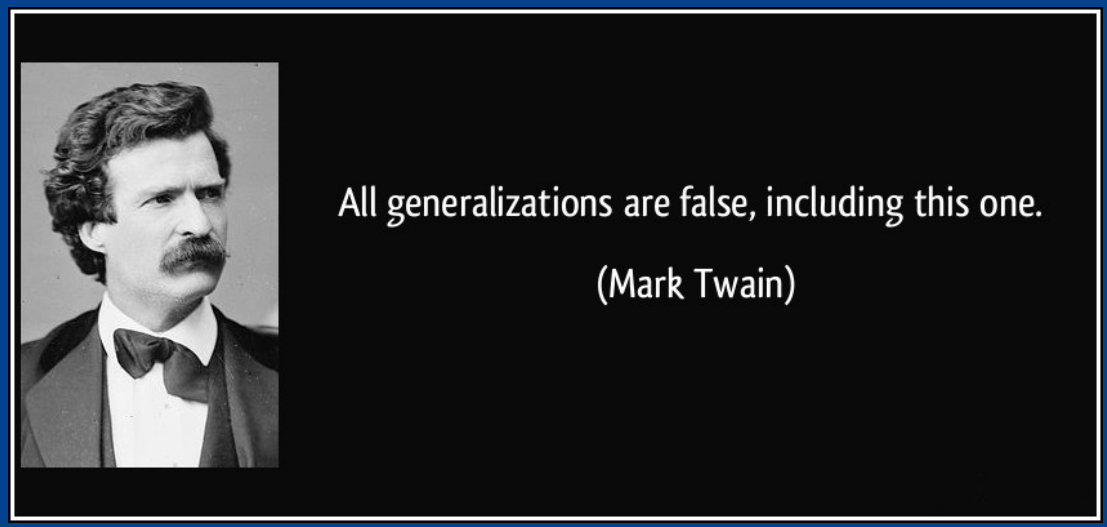
k-NN from L01:




$\Longrightarrow$ how to choose the **optimal** capacity?

NOTE: HOML: *Constraining a model [..] reduce risk of overfitting [via]* regularization => L08

# GENERALIZATION ERROR

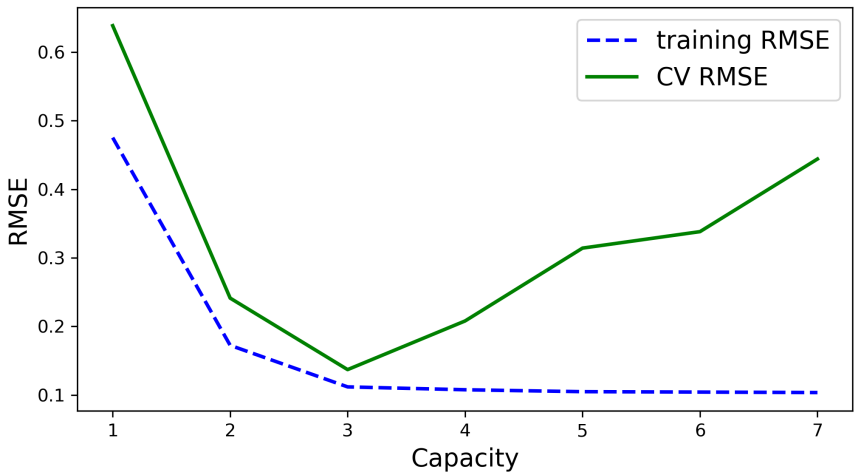


# Generalization Error

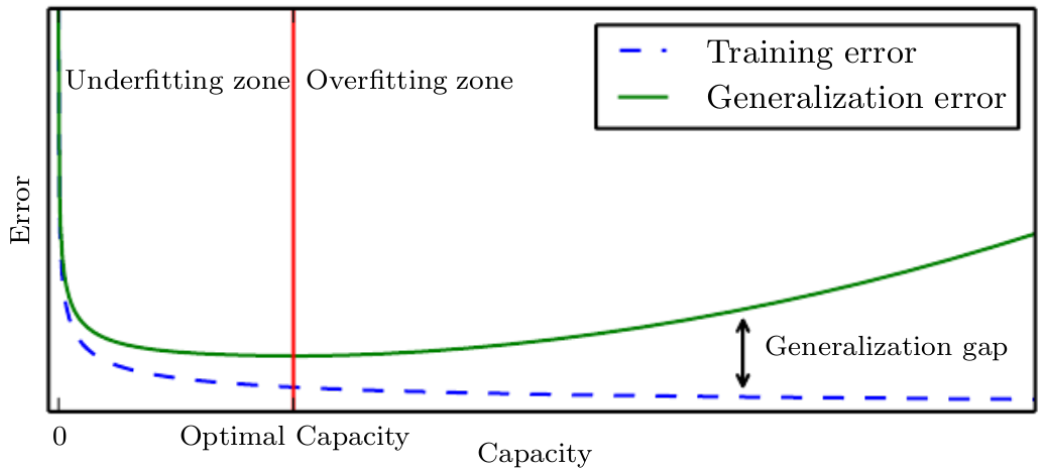
## Exercise: `generalization_error.ipynb`

RMSE-capacity plot for lin. reg. with polynomial features

(capacity $\sim$ degree of poly)




(Figure 5.3 from [DL])




Inspecting the plots from the exercise (`.ipynb`) and [DL], extracting the concepts:

- training/generalization error,
- generalization gab,
- underfit/overfit zone,
- optimal capacity (best-model, early stop),
- (and the two axes: x/capacity, y/error.)

# Generalization Error

Definition of ML:

> "A computer program is said to learn from experience $E$ with respect to some class of tasks $T$ and performance measure $P$ , if its performance at tasks in $T$, as measured by $P$, improves with experience $E$."
>
> — Mitchell (1997).

# Generalization Error

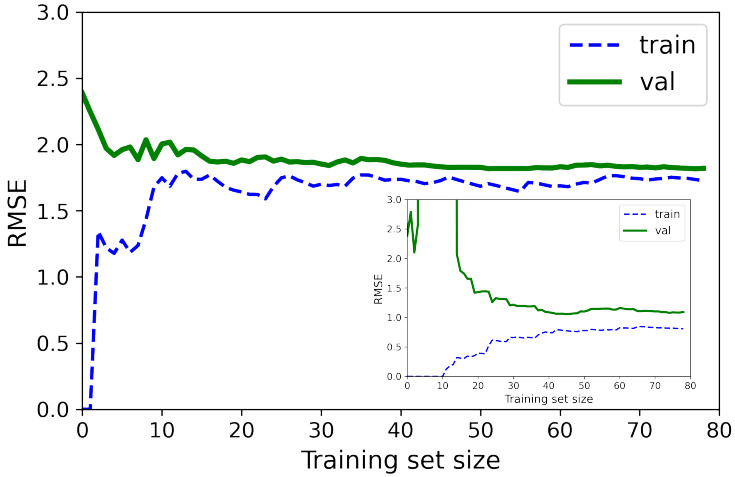
Exercise: `generalization_error.ipynb`

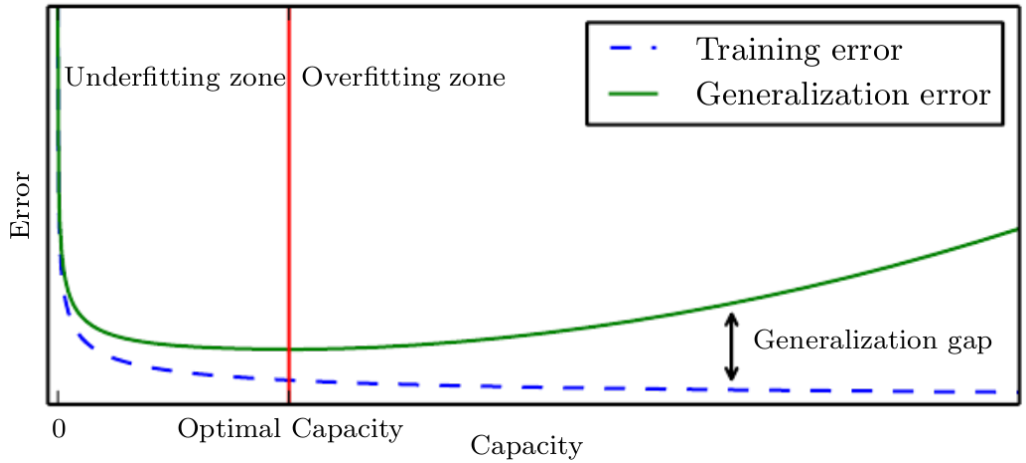
NOTE: three methods/plots:

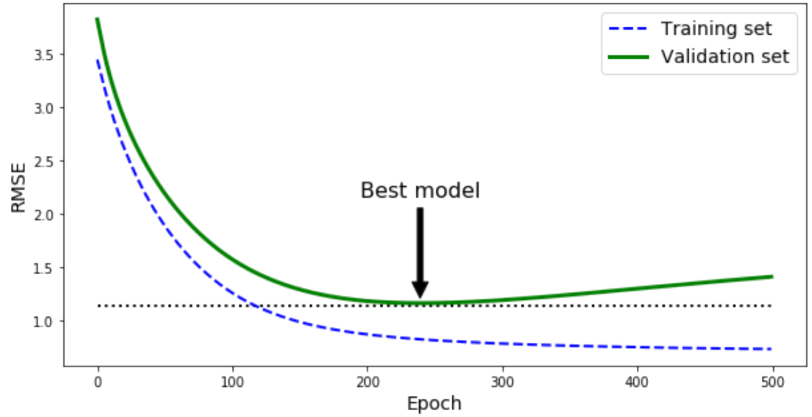
i) via **learning curves** as in [HOML],

ii) via an **error-capacity** plot as in [GITHOML] and [DL],

iii) via an **error-epoch** plot as in [GITHOML].



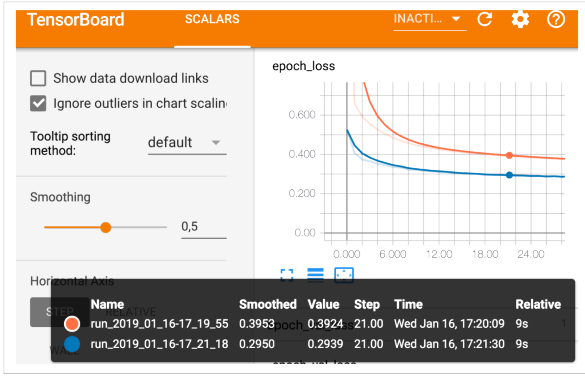




*Figure 10-16. Visualizing Learning Curves with TensorBoard*