ASSIGNMENT 2: GAUSSIAN CLASSIFIER, BIAS-VARIANCE, EVALUATION METRICS

Johannes Kofler

JOHANNES KEPLER UNIVERSITY LINZ

Institute for Machine Learning

Copyright statement:

Recap of formal framework: Model and loss function

- How do we get the "best" model?
  1. How does our model perform on our data? – Loss function
  2. How will it perform on (unseen) future data? (=how will it generalize?) – Generalization error/risk
- Assume we have a model $g(\mathbf{x}; \mathbf{w})$, parameterized by $\mathbf{w}$
- Its output should be as close as possible to the true target value y
- We use a loss function

$$L(y, g(\mathbf{x}; \mathbf{w}))$$

  to measure how close our prediction is to the true target.

Recap of formal framework: Generalization error/risk and Empirical Risk Minimization

- The generalization error or risk is the expected loss on future data:

$$R(g(.;\mathbf{w})) = \int\limits_X \int\limits_{\mathbb{R}} L(y, g(\mathbf{x};\mathbf{w}))\, p(\mathbf{x}, y)\, \mathrm{d}y\, \mathrm{d}\mathbf{x}$$

- In practice, we hardly have any knowledge about $p(\mathbf{x}, y)$. Precise definition: next slide.
- In practise: minimize the empirical risk R_{emp} on our dataset (Empirical Risk Minimization):

$$R_{\text{emp}}(g(.;\mathbf{w}), \mathbf{Z}_n) = \frac{1}{n} \sum_{i=1}^{n} L(y_i, g(\mathbf{x}_i;\mathbf{w}))$$

The probabilistic framework: Part 1

- Previous slide: assume that future data are generated according to joint distribution of inputs and outputs.
- The joint density (probability distribution) is denoted as $p(\mathbf{z}) = p(\mathbf{x}, y)$.
- If we have only finitely many possible data samples: $p(\mathbf{z})$ becomes probability to observe $\mathbf{z}$.
- Further important probabilistic objects in this context:
  1. Marginal distributions:
     - $p(\mathbf{x})$: density/probability of observing input vector $\mathbf{x}$ (regardless of target value)
     - $p(y)$: density/probability of observing target value y
  2. Conditional distributions:
     - $p(\mathbf{x} \mid y)$: density/probability of input value $\mathbf{x}$ for a given y
     - $p(y \mid \mathbf{x})$: density/probability to observe y for a given input $\mathbf{x}$

The probabilistic framework: Part 2

- By definition of conditional probability:

$$p(\mathbf{x}, y) = p(\mathbf{x}\,|\,y)\,p(y)$$
$$p(\mathbf{x}, y) = p(y\,|\,\mathbf{x})\,p(\mathbf{x})$$

- Bayes' Theorem:

$$p(y\,|\,\mathbf{x}) = \frac{p(\mathbf{x}|y)\,p(y)}{p(\mathbf{x})}, \quad p(\mathbf{x}\,|\,y) = \frac{p(y|\mathbf{x})\,p(\mathbf{x})}{p(y)}$$

- Marginal densities are obtained by integrating out:

$$p(\mathbf{x}) = \int\limits_{\mathbb{R}} p(\mathbf{x}, y)\,\mathrm{d}y = \int\limits_{\mathbb{R}} p(\mathbf{x}\,|\,y)\,p(y)\,\mathrm{d}y$$
$$p(y) = \int\limits_{X} p(\mathbf{x}, y)\,\mathrm{d}\mathbf{x} = \int\limits_{X} p(y\,|\,\mathbf{x})\,p(\mathbf{x})\,\mathrm{d}\mathbf{x}$$

- Next slides: use these concepts to provide an example where g can be calculated explicitly

Binary classification with 0-1 loss: Part 1

- Recall 0-1 loss: $L_{\mathbf{zo}}(y, g(\mathbf{x}; \mathbf{w})) = \begin{cases} 0 & y = g(\mathbf{x}; \mathbf{w}) \\ 1 & y \neq g(\mathbf{x}; \mathbf{w}) \end{cases}$
- Inserting this into the general formula of the risk, we obtain:

$$R(g(.; \mathbf{w})) = \int\limits_X \int\limits_{\mathbb{R}} p(\mathbf{x}, y \neq g(\mathbf{x}; \mathbf{w})) \, \mathrm{d}y \, \mathrm{d}\mathbf{x},$$

  i.e. the misclassification probability.
- Now we use binary classification with only two possible labels $y = \pm 1$. Then $\int \mathrm{d}y \to \sum_{y=\pm 1}$ and

$$R(g(.; \mathbf{w})) = \int\limits_X \sum_{y=\pm 1} p(\mathbf{x}, y \neq g(\mathbf{x}; \mathbf{w})) \, \mathrm{d}\mathbf{x}$$

Binary classification with 0-1 loss: Part 2

- Together with definition of conditional probability:

$$R(g(.;\mathbf{w})) = \int_X \left\{ \begin{array}{ll} p(\mathbf{x}, y=-1) & \text{if } g(\mathbf{x};\mathbf{w}) = +1 \\ p(\mathbf{x}, y=+1) & \text{if } g(\mathbf{x};\mathbf{w}) = -1 \end{array} \right\} \mathrm{d}\mathbf{x}$$

$$= \int_X \left\{ \begin{array}{ll} p(y=-1 \mid \mathbf{x}) & \text{if } g(\mathbf{x};\mathbf{w}) = +1 \\ p(y=+1 \mid \mathbf{x}) & \text{if } g(\mathbf{x};\mathbf{w}) = -1 \end{array} \right\} p(\mathbf{x})\,\mathrm{d}\mathbf{x}$$

- Optimal classifier (so-called Bayes-optimal classifier):

$$g_{\mathbf{opt}}(\mathbf{x}) = \left\{ \begin{array}{ll} +1 & \text{if } p(y=+1 \mid \mathbf{x}) \geq p(y=-1 \mid \mathbf{x}) \\ -1 & \text{if } p(y=-1 \mid \mathbf{x}) > p(y=+1 \mid \mathbf{x}) \end{array} \right\}$$

$$= \text{sign}(p(y=+1 \mid \mathbf{x}) - p(y=-1 \mid \mathbf{x}))$$

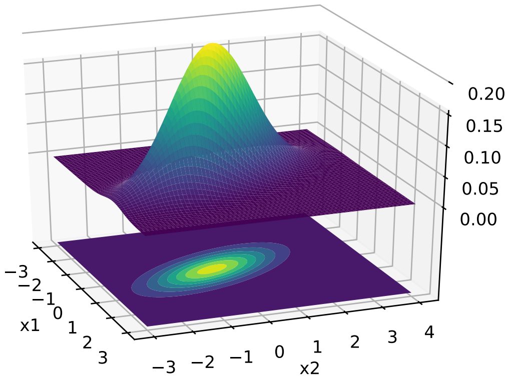
- Resulting minimal risk:

$$R_{\min} = \int_X \min(p(y=-1 \mid \mathbf{x}), p(y=+1 \mid \mathbf{x}))\, p(\mathbf{x})\,\mathrm{d}\mathbf{x}$$

- Next: formulas for Gauss distributed $p(\mathbf{x} \mid y = \pm 1)$

Interlude: intuition for multivariate Gauss distribution: Part 5

2-d Gaussian density with $\boldsymbol{\mu} = \begin{pmatrix} -1 \\ 0 \end{pmatrix}$, $\boldsymbol{\Sigma} = \begin{pmatrix} \frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & \frac{3}{2} \end{pmatrix}$



Explicit example: Gaussian classifier: Part 1

- We assume:

$$p(\mathbf{x} \mid y = -1) \sim N(\boldsymbol{\mu}_{-1}, \boldsymbol{\Sigma}_{-1})$$
$$p(\mathbf{x} \mid y = +1) \sim N(\boldsymbol{\mu}_{+1}, \boldsymbol{\Sigma}_{+1})$$

- Moreover we set:

$$\begin{aligned}
\bar{g}(\mathbf{x}) &= p(y = +1 \mid \mathbf{x}) - p(y = -1 \mid \mathbf{x}) \\
&= \frac{1}{p(\mathbf{x})} \left[p(\mathbf{x} \mid y = +1)\, p(y = +1) \right. \\
&\qquad \left. - \, p(\mathbf{x} \mid y = -1)\, p(y = -1) \right] \\
\hat{g}(\mathbf{x}) &= \ln p(\mathbf{x} \mid y = +1) + \ln p(y = +1) \\
&\qquad - \ln p(\mathbf{x} \mid y = -1) - \ln p(y = -1)
\end{aligned}$$

Explicit example: Gaussian classifier: Part 2

- Using the formula for $g_{\mathbf{opt}}$, we can infer

$$g_{\mathbf{opt}}(\mathbf{x}) = \text{sign}(\bar{g}(\mathbf{x})) = \text{sign}(\hat{g}(\mathbf{x}))$$

- Plugging in the definitions: optimal classification border $\hat{g}(\mathbf{x}) = 0$ is d-dimensional hyper-quadric (without proof):

$$-\tfrac{1}{2}\mathbf{x}^T\mathbf{A}\mathbf{x} + \mathbf{b}^T\mathbf{x} + c = 0.$$

- Where:
  1. $\mathbf{A} = \boldsymbol{\Sigma}_{+1}^{-1} - \boldsymbol{\Sigma}_{-1}^{-1}$
  2. $\mathbf{b} = \boldsymbol{\Sigma}_{+1}^{-1}\boldsymbol{\mu}_{+1} - \boldsymbol{\Sigma}_{-1}^{-1}\boldsymbol{\mu}_{-1}$
  3. $c = -\frac{1}{2}\boldsymbol{\mu}_{+1}^T\boldsymbol{\Sigma}_{+1}^{-1}\boldsymbol{\mu}_{+1} + \frac{1}{2}\boldsymbol{\mu}_{-1}^T\boldsymbol{\Sigma}_{-1}^{-1}\boldsymbol{\mu}_{-1} - \frac{1}{2}\ln\det\boldsymbol{\Sigma}_{+1} + \frac{1}{2}\ln\det\boldsymbol{\Sigma}_{-1} + \ln p(y=+1) - \ln p(y=-1)$

- Next slides: provide visualizations

Explicit example: Gaussian classifier: Part 3: Concrete Example Assumptions

- $p(\boldsymbol{x} \mid y = +1) \sim N(\boldsymbol{\mu}_{+1}, \boldsymbol{\Sigma}_{+1})$ with:

$$\boldsymbol{\mu}_{+1} = (0.4, 0.8) \qquad \boldsymbol{\Sigma}_{+1} \approx \begin{pmatrix} 0.1 & 0.0 \\ 0.0 & 0.005 \end{pmatrix}$$

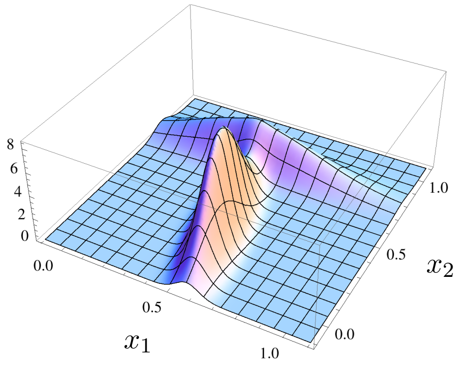
- $p(\mathbf{x} \mid y = -1) \sim N(\mu_{-1}, \boldsymbol{\Sigma}_{-1})$ with:

$$\boldsymbol{\mu}_{-1} = (0.5, 0.3) \qquad \boldsymbol{\Sigma}_{-1} \approx \begin{pmatrix} 0.004 & -0.007 \\ -0.007 & 0.04 \end{pmatrix}$$

- $p(y = +1) = \frac{55}{120} \approx 0.46,\ p(y = -1) = \frac{65}{120} \approx 0.54$

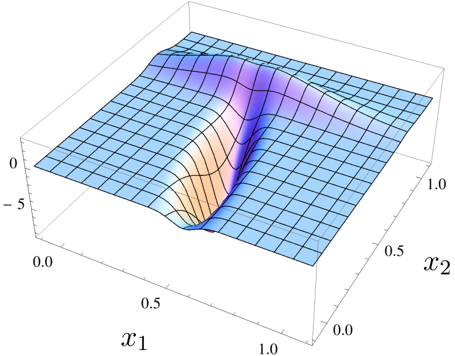
Explicit example: Gaussian classifier: Part 4: Density plot

$$p(\mathbf{x}) = p(\mathbf{x} \mid y = -1) \cdot p(y = -1) + p(\mathbf{x} \mid y = +1) \cdot p(y = +1)$$



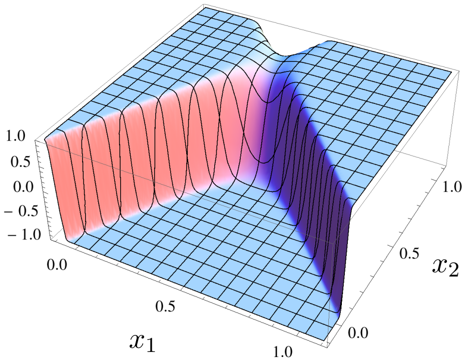
Explicit example: Gaussian classifier: Part 5: Plot of $\tilde{g}$

$$\tilde{g}(\mathbf{x}) = p(\mathbf{x} \mid y = +1) \cdot p(y = +1) - p(\mathbf{x} \mid y = -1) \cdot p(y = -1)$$

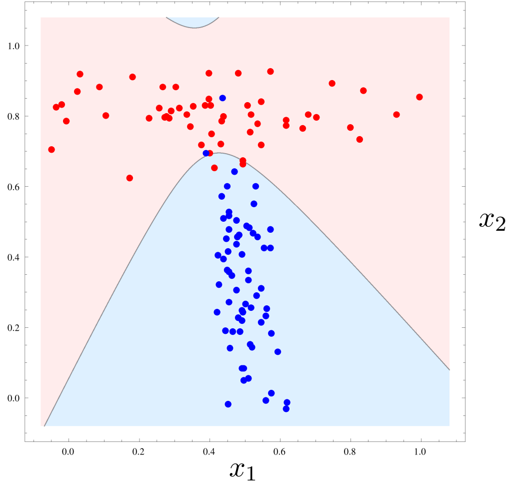


Explicit example: Gaussian classifier: Part 6: Plot of Discriminant function $\bar{g}$

$\bar{g}(\mathbf{x}) = \tilde{g}(\mathbf{x})/p(\mathbf{x})$



Explicit example: Gaussian classifier: Part 7: Plot of Data and Decision Boundary



Another explicit example: Linear regression in $d = 1$: Part 1: Basics

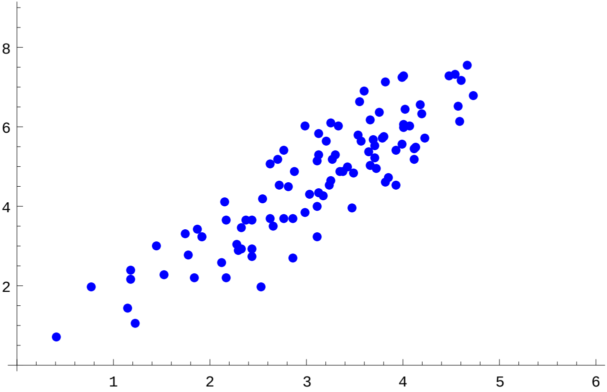
- Main ingredients:
  1. Dataset $\mathbf{Z} = \{(x_i, y_i) \mid i = 1, \ldots, l\}$ with $x_i, y_i \in \mathbb{R}$
  2. Linear classifier: $g(x; w_0, w_1) = w_0 + w_1\, x$
  3. Averaged quadratic loss:

$$
\begin{aligned}
Q(\mathbf{Z}; w_0, w_1) &= \frac{1}{l} \sum_{i=1}^{l} L_{\mathbf{q}}(y_i, g(x_i; w_0, w_1)) \\
&= \frac{1}{l} \sum_{i=1}^{l} \left(w_0 + w_1\, x_i - y_i\right)^2
\end{aligned}
$$

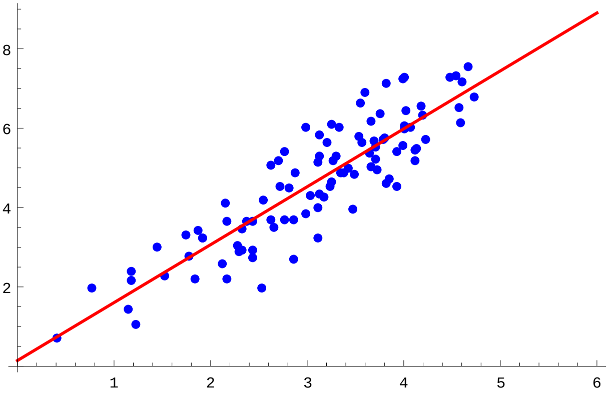
- Aim: Find solution (w_0, w_1) that minimizes $Q(\mathbf{Z}; w_0, w_1)$. Techniques from calculus and linear algebra lead to explicit formula
- For more details and intuitions be patient until Unit 5 or consider e.g. the course Basic Methods of Data Analysis.

Linear regression in $d = 1$: Part 2: Plot of Data

All subsequent plots are y versus x.



Linear regression in $d = 1$: Part 3: Plot of Data + Regression Line



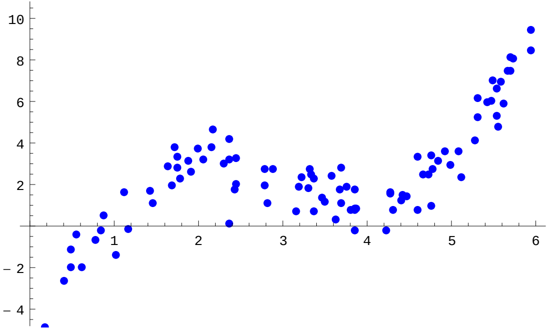
Polynomial regression in $d = 1$: Part 1: Basics

- For more complex data: more complex models; try polynomials
- Main ingredients:
  1. Dataset $\mathbf{Z} = \{(x_i, y_i) \mid i = 1, \ldots, l\}$ with $x_i, y_i \in \mathbb{R}$
  2. Polynomial classifier of degree m:
     $g(x; w_0, w_1, \ldots, w_m) = w_0 + w_1\, x + w_2\, x^2 + \cdots + w_m\, x^m$
  3. Averaged quadratic loss
- Again, there exists a unique global solution with an explicit formula:

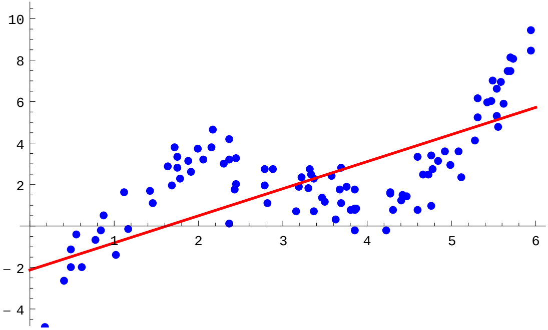
  $$\mathbf{w} = \left(\tilde{\mathbf{X}}^T\,\tilde{\mathbf{X}}\right)^{-1}\tilde{\mathbf{X}}^T\,\mathbf{y} \quad \text{with} \quad \tilde{\mathbf{X}} = (\mathbf{1}, \mathbf{x}, \mathbf{x}^{[2]}, \ldots, \mathbf{x}^{[m]})$$

  The design matrix $\tilde{\mathbf{X}}$ is a Vandermonde matrix.

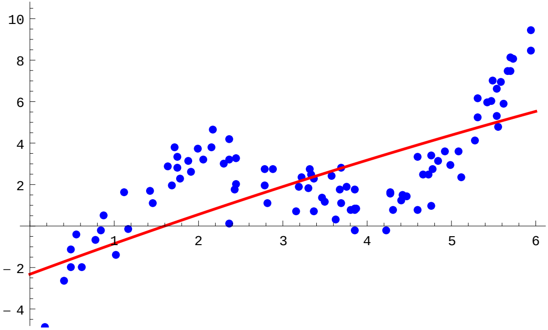
Polynomial regression in $d = 1$: Part 2: Plot of data



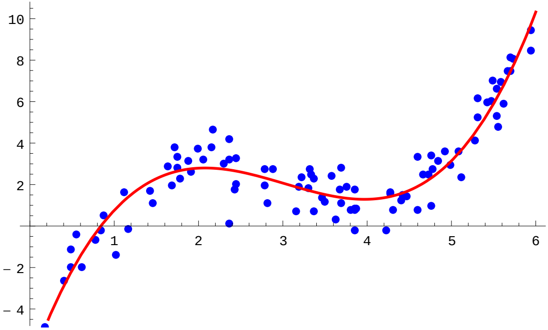
Polynomial regression in $d = 1$: Part 3: Regression with degree $m = 1$



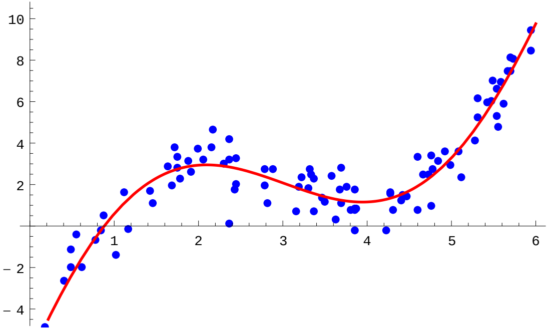
Polynomial regression in $d = 1$: Part 4: Regression with degree $m = 2$



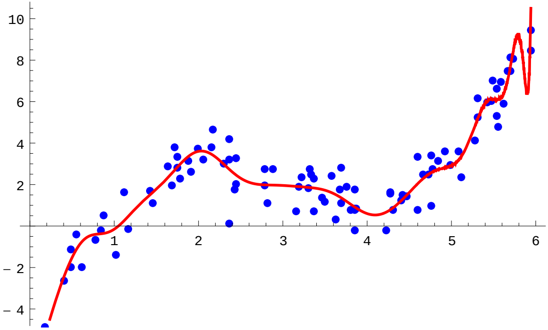
Polynomial regression in $d = 1$: Part 5: Regression with degree $m = 3$



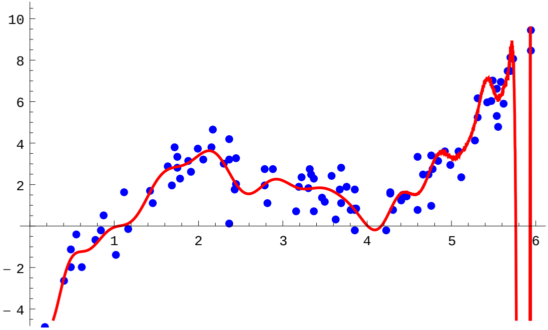
Polynomial regression in $d = 1$: Part 6: Regression with degree $m = 5$



Polynomial regression in $d = 1$: Part 7: Regression with degree $m = 25$



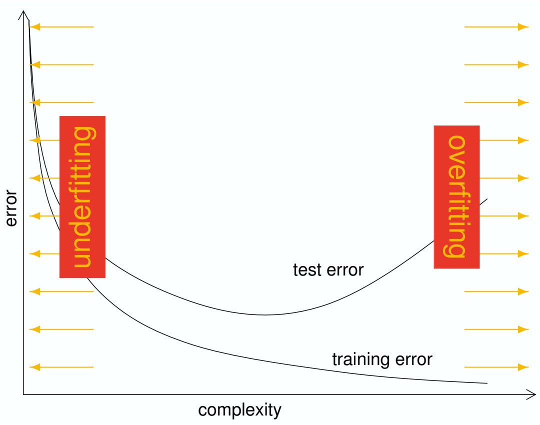
Polynomial regression in $d = 1$: Part 8: Regression with degree $m = 75$



Bias-Variance Tradeoff: Part 1: Intuition

- Previous example: instance of one of the basic problems of supervised machine learning: bias-variance tradeoff.
- Recall from Unit 1:
  1. Underfitting: model is too coarse to fit training or test data (too low model class complexity): e.g. $m = 1$
  2. Overfitting: model fits well to training data but not to future/test data (too high model class complexity): e.g. $m = 75$
- This rather general situation that often occurs in practice is illustrated in the next slides.
- We will also discuss these issues in more detail and on a more formal level.

Bias-Variance Tradeoff: Part 2: Notorious situation in practice



- Next slides: Explicit example of quadratic loss, where a nice decomposition with proper interpretation is possible

Bias-Variance Decomposition for Quadratic Loss: Part 1

- $\mathbf{Z}_l$: sample set of l elements
- Object of interest: expected prediction error (EPE) for $\mathbf{x}_0 \in X$:

$$\begin{aligned}\text{EPE}(\mathbf{x}_0) &= \mathrm{E}_{y|\mathbf{x}_0,\mathbf{Z}_l}\left[L_\mathbf{q}(y, g(\mathbf{x}_0; \mathbf{w}(\mathbf{Z}_l)))\right] \\ &= \mathrm{E}_{y|\mathbf{x}_0,\mathbf{Z}_l}\left[(y - g(\mathbf{x}_0; \mathbf{w}(\mathbf{Z}_l)))^2\right]\end{aligned}$$

- By assumption: $y \,|\, \mathbf{x}_0$ and the selection of training samples are independent, thus:

$$\text{EPE}(\mathbf{x}_0) = \mathrm{E}_{y|\mathbf{x}_0}\left[\mathrm{E}_{\mathbf{Z}_l}\left[(y - g(\mathbf{x}_0; \mathbf{w}(\mathbf{Z}_l)))^2\right]\right]$$

- A short calculation yields (see exercises):

$$\begin{aligned}\text{EPE}(\mathbf{x}_0) = &\,\mathrm{Var}[y \,|\, \mathbf{x}_0] \\ &+ \Big(\mathrm{E}[y \,|\, \mathbf{x}_0] - \mathrm{E}_{\mathbf{Z}_l}\left[g(\mathbf{x}_0; \mathbf{w}(\mathbf{Z}_l))\right]\Big)^2 \\ &+ \mathrm{E}_{\mathbf{Z}_l}\left[\Big(g(\mathbf{x}_0; \mathbf{w}(\mathbf{Z}_l)) - \mathrm{E}_{\mathbf{Z}_l}[g(\mathbf{x}_0; \mathbf{w}(\mathbf{Z}_l))]\Big)^2\right]\end{aligned}$$

Bias-Variance Decomposition for Quadratic Loss: Part 2

1. The first term

$$\mathrm{Var}[y \,|\, \mathbf{x}_0]$$

   measures the label variance, i.e. the amount to which the label y varies at $\mathbf{x}_0$: unavoidable error.

2. The second term

$$\text{bias}^2 = \Big(\mathrm{E}[y \,|\, \mathbf{x}_0] - \mathrm{E}_{\mathbf{Z}_l}\big[g(\mathbf{x}_0; \mathbf{w}(\mathbf{Z}_l))\big]\Big)^2$$

   measures how close the model in average approximates the average target y at $\mathbf{x}_0$: squared bias.

3. The third term,

$$\text{variance} = \mathrm{E}_{\mathbf{Z}_l}\Big[\big(g(\mathbf{x}_0; \mathbf{w}(\mathbf{Z}_l)) - \mathrm{E}_{\mathbf{Z}_l}[g(\mathbf{x}_0; \mathbf{w}(\mathbf{Z}_l))]\big)^2\Big]$$

   is the variance of the model at $\mathbf{x}_0$, i.e. $\mathrm{Var}_{\mathbf{Z}_l}[g(\mathbf{x}_0; \mathbf{w}(\mathbf{Z}_l))]$.

The Bias-Variance Trade-off: Summary

- Minimizing the generalization error (learning) is concerned with optimizing bias and variance simultaneously.
- Underfitting $=$ high bias $=$ too simple model
- Overfitting $=$ high variance $=$ too complex model
- Empirical risk minimization does not include any mechanism to assess bias and variance independently (how should it?)
- More specifically: if we do not care about model complexity (in particular, if we allow highly or even arbitrarily complex models), ERM has high risk to produce over-fitted models.

Evaluation of classifiers: possible pitfalls

- So far: only performance measure was generalization error based on $L_{\mathbf{zo}}$
- Another frequent problem mentioned already in Unit 1: unbalanced data sets
- What if misclassification cost depends on the sample's class?
- Can we define a general performance measure independent of class distributions and misclassification costs?
- To answer these questions: introduce confusion matrices

Confusion matrix for binary classification: Part 1

For a given sample $(\mathbf{x}, y)$ and a classifier $g(.)$): $(\mathbf{x}, y)$ is a

- true positive (TP) if $y = +1$ and $g(\mathbf{x}) = +1$ (hit),
- true negative (TN) if $y = -1$ and $g(\mathbf{x}) = -1$ (correct rejection),
- false positive (FP) if $y = -1$ and $g(\mathbf{x}) = +1$ (false alarm),
- false negative (FN) if $y = +1$ and $g(\mathbf{x}) = -1$ (miss).

Confusion matrix for binary classification: Part 2

Given a data set $(\mathbf{z}^1, \ldots, \mathbf{z}^m)$, the confusion matrix is defined as follows:

		predicted value $g(\mathbf{x};\mathbf{w})$	
		$+1$	-1
actual value y	$+1$	#TP	#FN
	-1	#FP	#TN

The entries #TP, #FP, #FN and #TN denote the numbers of true positives, . . . , respectively, for the given data set.

Evaluation measures derived from confusion matrix

- Positives: #P $=$ #TP$+$#FN
- Negatives: #N $=$ #TN$+$#FP
- Accuracy: proportion of correctly classified items: ACC $= \frac{\#TP+\#TN}{\#TP+\#TN+\#FP+\#FN}$.
- True Positive Rate (aka Recall, Sensitivity): proportion of correctly identified positives: TPR $= \frac{\#TP}{\#TP+\#FN}$.
- False Positive Rate: proportion of negative examples that were incorrectly classified as positives: FPR $= \frac{\#FP}{\#TN+\#FP}$.
- Precision: proportion of predicted positive examples that were correct: PREC $= \frac{\#TP}{\#TP+\#FP}$.
- True Negative Rate (aka Specificity): proportion of correctly identified negatives: TNR $= \frac{\#TN}{\#TN+\#FP}$.
- False Negative Rate: proportion of positive examples that were incorrectly classified as negatives: FNR $= \frac{\#FN}{\#TP+\#FN}$.

Evaluation measures for unbalanced data

- Balanced Accuracy: mean of true positive and true negative rate, i.e.

$$\text{BACC} = \frac{\text{TPR}+\text{TNR}}{2}$$

- Matthews Correlation Coefficient: measure of non-randomness of classification; defined as normalized determinant of confusion matrix, i.e.

$$\text{MCC} = \frac{\#\text{TP}\cdot\#\text{TN}-\#\text{FP}\cdot\#\text{FN}}{\sqrt{(\#\text{TP}+\#\text{FP})(\#\text{TP}+\#\text{FN})(\#\text{TN}+\#\text{FP})(\#\text{TN}+\#\text{FN})}}$$

- F-score: harmonic mean of precision and recall, i.e.

$$F_1 = 2\cdot\frac{\text{PREC}\cdot\text{TPR}}{\text{PREC}+\text{TPR}}$$

- Next: generalize previously introduced concepts to multi-class classification