STAT 593
Robust statistics:
Modeling and Computing

Joseph Salmon
http://josephsalmon.eu

Télécom Paristech, Institut Mines-Télécom
&
University of Washington, Department of Statistics
(Visiting Assistant Professor)

Outline

Table of Contents

Presentation

Joseph Salmon (Assistant Professor):

- Positions:
  - PhD. student at Paris Diderot-Paris 7 (2007-2010)
  - Post-Doc at Duke University (2011-2012)
  - Assistant Professor at Télécom ParisTech (2012-)
  - Visiting Assistant Professor at UW (2018)
- Research themes: high dimensional statistics, optimization for machine learning, aggregation, image processing
- Email: *joseph.salmon@telecom-paristech.fr*
- Website: *josephsalmon.eu*

(No) Grades / office hours

Beware: this is a **Credit/No-Credit** grading course

- Office hours : Friday 10:30-11:30 AM; **by appointment only**
- Office: B314 Padelford
- Number of credits: 3

Outline of the course

Week 1 Introduction, examples, basic concepts, location, scale, equivariance

Week 2 Breaking point, M-estimates, pseudo-observations

Week 3 L-statistics: Linear combination of order statistics

Week 4 Numerical computation of M-estimates, non-smooth convex optimization, Iterative Re-weighted Least Square IRLS

Week 5 Smoothing non smooth problems

Week 6 Gâteaux differentiability, Sensitivity curve, Influence Function

Week 7 Robust regression and multivariate statistics

Week 8 Quantile regression, “crossing”

Week 9 Guest Lectures

Week 10 Project presentations

Table of Contents

Prerequisites

- **Probability** basis: probability, expectation, law of large number, Gaussian distribution, central limit theorem.
  Books: Murphy (2012, ch.1 and 2)
- **Optimisation** basis: (differential) calculus, convexity, first order conditions, gradient descent, Newton method
  Books: Boyd and Vandenberghe (2004), Bertsekas (1999)

Prerequisites

- **Probability** basis: probability, expectation, law of large number, Gaussian distribution, central limit theorem.
  Books: Murphy (2012, ch.1 and 2)
- **Optimisation** basis: (differential) calculus, convexity, first order conditions, gradient descent, Newton method
  Books: Boyd and Vandenberghe (2004), Bertsekas (1999)
- **(bi-)linear algebra** basis: vector space, norms, inner product, matrices, determinants, diagonalization
  Lecture: Horn and Johnson (1994)

Prerequisites

- **Probability** basis: probability, expectation, law of large number, Gaussian distribution, central limit theorem.
  Books: Murphy (2012, ch.1 and 2)
- **Optimisation** basis: (differential) calculus, convexity, first order conditions, gradient descent, Newton method
  Books: Boyd and Vandenberghe (2004), Bertsekas (1999)
- **(bi-)linear algebra** basis: vector space, norms, inner product, matrices, determinants, diagonalization
  Lecture: Horn and Johnson (1994)
- **Numerical linear algebra**: linear system resolution, Gaussian elimination, matrix factorization, conditioning, etc.
  Lecture: Golub and VanLoan (2013), Applied Numerical Computing by L. Vandenberghe

Prerequisites

- **Probability** basis: probability, expectation, law of large number, Gaussian distribution, central limit theorem.
  Books: Murphy (2012, ch.1 and 2)
- **Optimisation** basis: (differential) calculus, convexity, first order conditions, gradient descent, Newton method
  Books: Boyd and Vandenberghe (2004), Bertsekas (1999)
- **(bi-)linear algebra** basis: vector space, norms, inner product, matrices, determinants, diagonalization
  Lecture: Horn and Johnson (1994)
- **Numerical linear algebra**: linear system resolution, Gaussian elimination, matrix factorization, conditioning, etc.
  Lecture: Golub and VanLoan (2013), Applied Numerical Computing by L. Vandenberghe

Books, recommended lectures

Books on robust statistics:

- Maronna *et al.* (2006)
- Huber and Ronchetti (2009)
- Hampel *et al.* (1986)
- Rousseeuw and Leroy (1987)

Book for linear models:

- Seber and Lee (2003)

Book for optimization, Legendre/Fenchel conjugacy:

- Hiriart-Urruty and Lemarechal (1993,1993b)
- Bauschke and Combettes (2011)

Surveys on optimization:

- Parikh *et al.* (2013)

Algorithmic aspects: some advice

`Python` installation: use **Conda / Anaconda**

Recommended tools: **Jupyter / IPython Notebook**, **IPython** with a text editor *e.g.*, **Atom**, **Sublime Text**, **Visual Studio Code**, etc.

- **Python**, **Scipy**, **Numpy**: https://jakevdp.github.io/PythonDataScienceHandbook/
- **Pandas**: http://pandas.pydata.org/
- **scikit-learn**: http://scikit-learn.org/stable/
- **Statmodels**: http://www.statsmodels.org

General advice

- Use version control system for your work:
  Git (*e.g.*, Bitbucket, Github, etc.) or **Mercurial**
- Use clean way of writing / presenting your code
  Example : **PEP8** for Python (use for instance **AutoPEP8**, `https://github.com/kenkoooo/jupyter-autopep8`)
- Learn from good examples:
  `https://github.com/scikit-learn/`,
  `http://jakevdp.github.io/`, etc.

List of interesting papers (I)

- Depth[1][2]
- Linear models / Lasso methods[3][4][5]

---

[1] D. L. Donoho and M. Gasko. "Breakdown properties of location estimates based on halfspace depth and projected outlyingness". In: *Ann. Statist.* 20.4 (1992), pp. 1803–1827.

[2] K. Mosler. "Depth statistics". In: *Robustness and complex data structures*. Springer, 2013, pp. 17–34.

[3] M. Avella-Medina and E. M. Ronchetti. "Robust and consistent variable selection in high-dimensional generalized linear models". In: *Biometrika* 105.1 (2018), pp. 31–44.

[4] H. Xu, C. Caramanis, and S. Mannor. "Robust regression and Lasso". In: *IEEE Trans. Inf. Theory* 56.7 (2010), pp. 3561–3574.

[5] A. Alfons, C. Croux, and S. Gelper. "Sparse least trimmed squares regression for analyzing high-dimensional large data sets". In: *Ann. Appl. Stat.* 7.1 (2013), pp. 226–248.

List of interesting papers (II)

- Robust optimization point of view[6][7]
- Robust covariance estimation[8]
- Geometric median[9][10]
- Smoothing non-smooth functions[11][12]

[6] Y. Chen, C. Caramanis, and S. Mannor. "Robust sparse regression under adversarial corruption". In: *ICML*. 2013, pp. 774–782.

[7] D. Bertsimas, D. B. Brown, and C. Caramanis. "Theory and applications of robust optimization". In: *SIAM Rev.* 53.3 (2011), pp. 464–501.

[8] M. Chen, C. Gao, and Z. Ren. "A General Decision Theory for Huber's ϵ-Contamination Model". In: *Electron. J. Stat.* 10.2 (2016), pp. 3752–3774.

[9] S. Minsker. "Geometric median and robust estimation in Banach spaces". In: *Bernoulli* 21.4 (2015), pp. 2308–2335.

[10] X. Wei and S. Minsker. "Estimation of the covariance structure of heavy-tailed distributions". In: *NIPS*. 2017, pp. 2859–2868.

[11] Y. Nesterov. "Smooth minimization of non-smooth functions". In: *Math. Program.* 103.1 (2005), pp. 127–152.

[12] A. Beck and M. Teboulle. "Smoothing and first order methods: A unified framework". In: *SIAM J. Optim.* 22.2 (2012), pp. 557–580.

Table of Contents

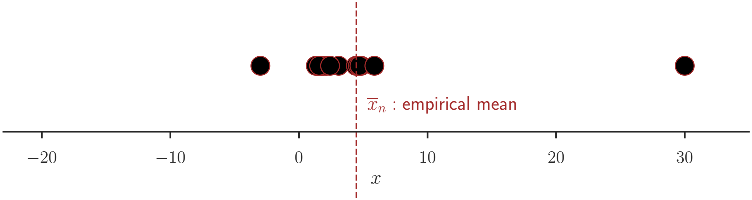
Notation / Settings

Observations: n samples $x_1, \ldots, x_n$ real numbers; later real vector will be elements of $\mathbb{R}^d$

Vector notation : n samples $x_1, \ldots, x_n$ (or $y_1, \ldots, y_n$):
$\mathbf{x} = (x_1, \ldots, x_n)^\top \in \mathbb{R}^n$ (or $\mathbf{y} = (y_1, \ldots, y_n)^\top \in \mathbb{R}^n$)

Inner product: $\langle \mathbf{x}, \mathbf{y} \rangle = \sum_{i=1}^{n} x_i y_i$

Sample Mean (empirical mean)

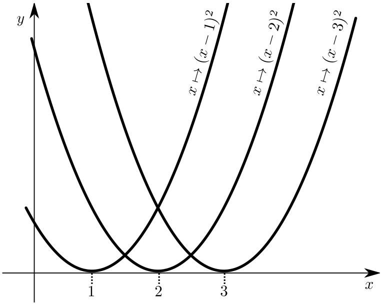


Definition

$$\textbf{Sample mean :}\quad \overline{x}_n = \frac{1}{n}\sum_{i=1}^{n} x_i = \mathop{\arg\min}_{\mu\in\mathbb{R}} \sum_{i=1}^{n}(\mu - x_i)^2$$

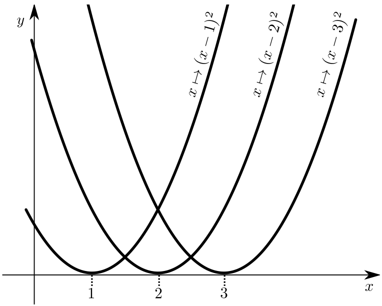
Rem: $\overline{x}_n = \langle \mathbf{x}, \frac{\mathbf{1}_n}{n} \rangle$, where $\mathbf{1}_n = (1, \ldots, 1)^\top \in \mathbb{R}^n$ and $\overline{x}_n \mathbf{1}_n$ is the (Euclidean) projection of $\mathbf{x}$ on $\mathrm{Span}(\mathbf{1}_n)$

Mean: optimization problem

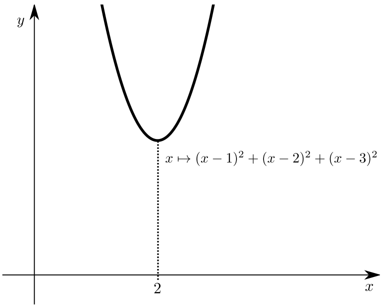


Individual objectives

Mean: optimization problem

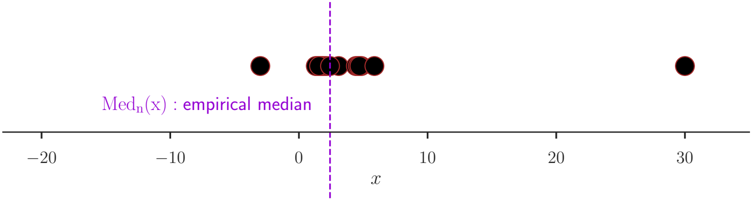


Individual objectives



Sum of individual objective

Median



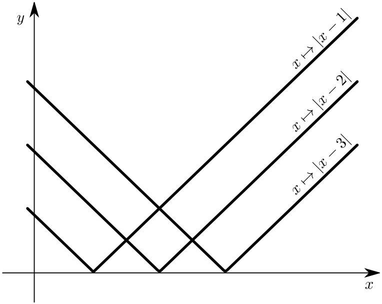
Definition

$$\textbf{Median}: \quad \mathrm{Med}_n(\mathbf{x}) \in \arg\min_{\mu \in \mathbb{R}} \sum_{i=1}^{n} |\mu - x_i|$$

Rem: often, with $x_{(1)} \leq x_{(2)} \leq \ldots \leq x_{(n)}$ (**order statistics**)

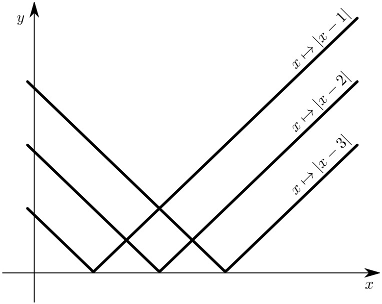
$$\mathrm{Med}_n(\mathbf{x}) = \begin{cases} \frac{x_{(\frac{n}{2})} + x_{(\frac{n}{2}+1)}}{2}, & \text{if } n \text{ is even} \\ x_{(\frac{n+1}{2})}, & \text{if } n \text{ is odd} \end{cases}$$

Median: optimization problem

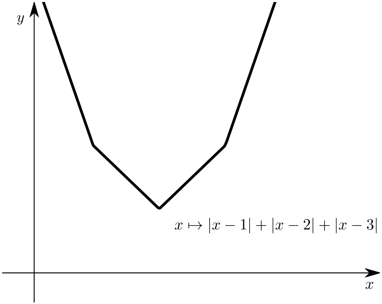


Individual objectives

Median: optimization problem

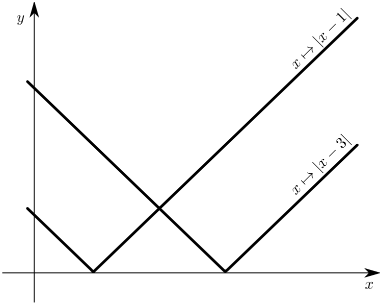


Individual objectives



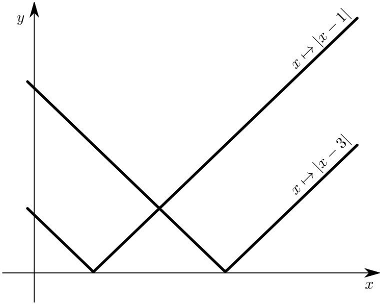
Sum of individual objective

Median: optimization problem

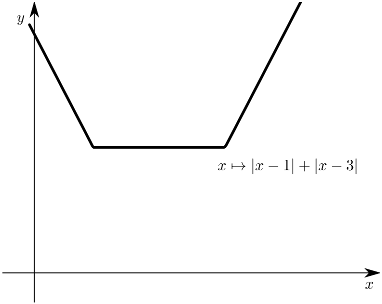


Individual objectives

Median: optimization problem

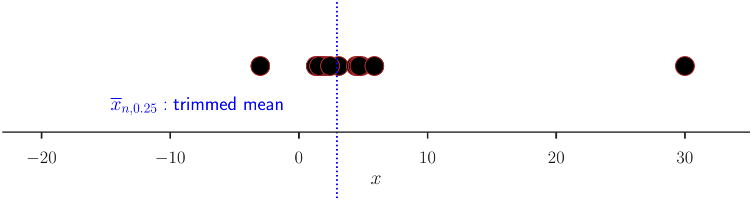


Individual objectives



Sum of individual objective

Trimmed mean



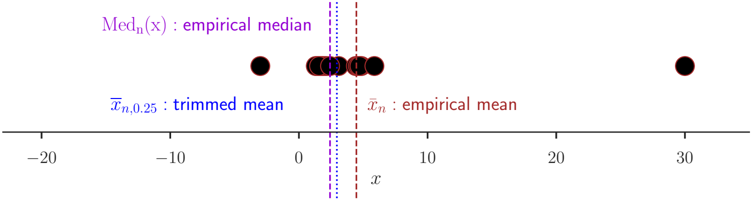
Definition

Trimmed mean (at level α) : $$\overline{x}_{n,\alpha} = \frac{1}{n-2m} \sum_{i=m+1}^{n-m} x_{(i)}$$

where $m = \lfloor (n-1)\alpha \rfloor$ and $x_{(i)}$ denotes the order statistics

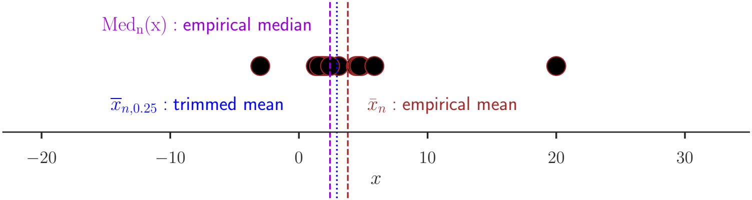
Rem: $\lfloor u \rfloor$ is the integer part of u

Mean *vs* median *vs* Trimmed Mean



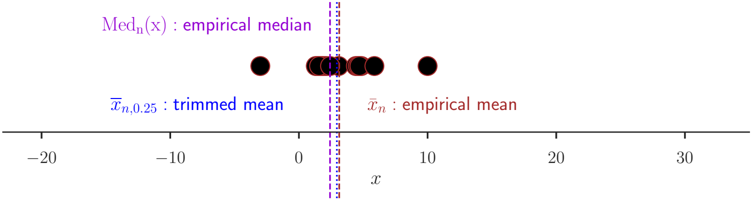
- Trimmed Mean and median are robust to outliers; the (empirical) mean is not

Mean *vs* median *vs* Trimmed Mean



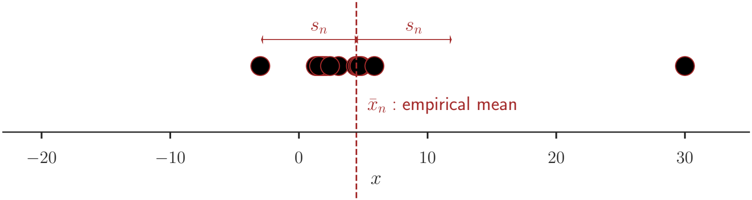
- Trimmed Mean and median are robust to outliers; the (empirical) mean is not

Mean *vs* median *vs* Trimmed Mean



- Trimmed Mean and median are robust to outliers; the (empirical) mean is not

Variance / standard-deviation



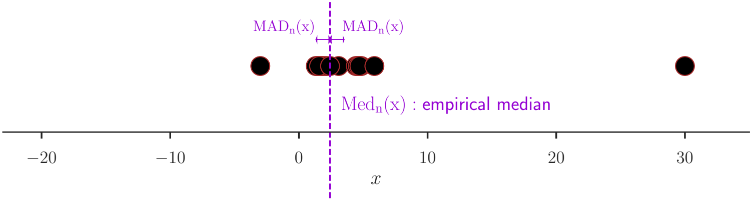
Definition

Variance : $$\mathrm{var}_n(\mathbf{x}) = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \overline{x}_n)^2 = \frac{1}{n-1}\left\|\mathbf{x} - \overline{x}_n \mathbf{1}_n\right\|^2$$

Std : $$s_n(\mathbf{x}) = \sqrt{\mathrm{var}_n(\mathbf{x})} \quad (\text{where } \|\mathbf{z}\|^2 = \sum_{i=1}^{n} z_i^2)$$

<u>Rem</u>: normalization can change $1/n$ or $1/(n-1)$ (unbiased)

Mean Absolute Deviation



Definition

Mean Absolute Deviation (MAD):

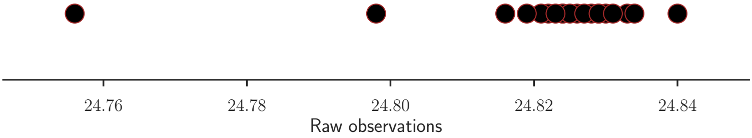
$$\mathrm{MAD}_n(\mathbf{x}) = \mathrm{Med}_n\left(|\mathrm{Med}_n(\mathbf{x}) - \mathbf{x}|\right)$$

Normalized Mean Absolute Deviation (MADN):

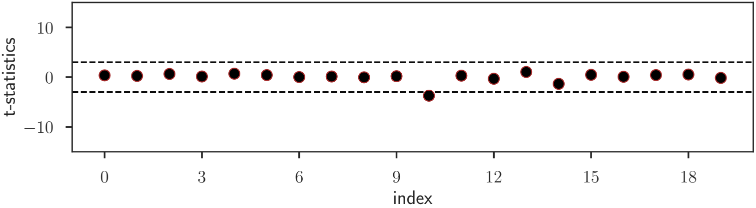
$$\mathrm{MADN}_n(\mathbf{x}) = \mathrm{MAD}_n(\mathbf{x})/0.6745$$

Rem: $\Phi^{-1}(3/4) \approx 0.6745$ (Φ : Standard Gaussian CDF)

Newcomb's experiments (speed of light)



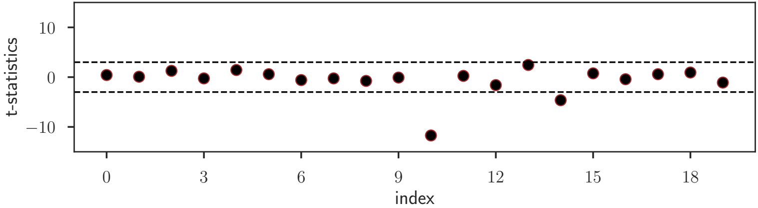
Newcomb's experiments (speed of light)



Standard statistical rule of thumb "3σ": flag a sample x_i as outlier when $|t_i| > 3$, where

$$t_i = \frac{x_i - \overline{x}_n}{s_n}$$

Newcomb's experiments (speed of light)



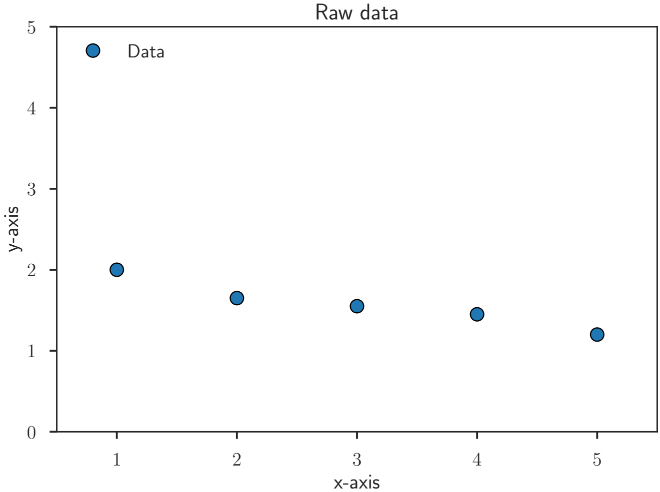
Robust counterpart for "3σ" rule of thumb: flag a sample x_i as outlier when $|t'_i| > 3$, where

$$t'_i = \frac{x_i - \mathrm{Med}_n(\mathbf{x})}{\mathrm{MADN}_n(\mathbf{x})}$$

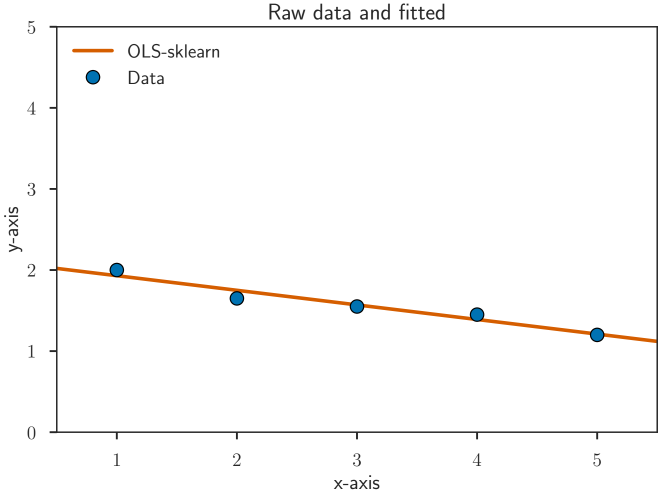
<u>Rem</u>: helps limiting the **masking** effect

Table of Contents

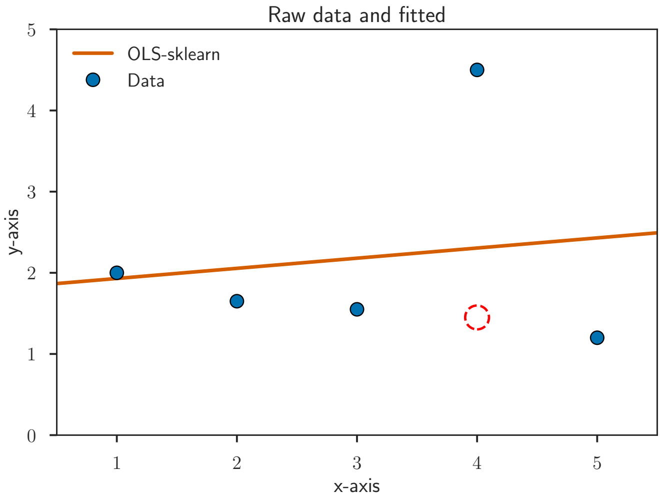
Ordinary Least Squares: toy example (y-corruption)



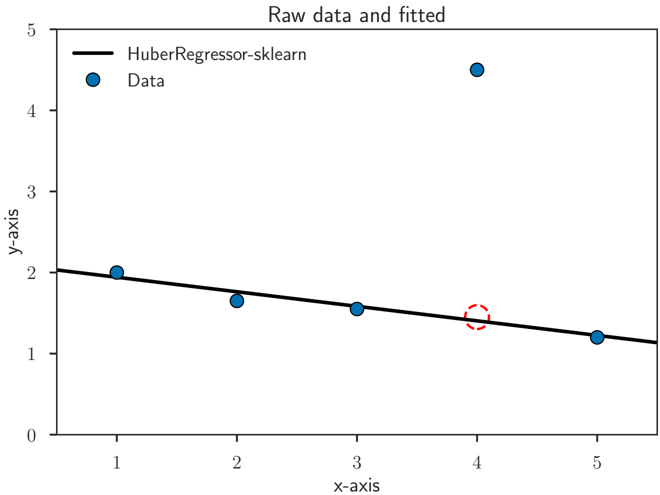
Ordinary Least Squares: toy example (y-corruption)



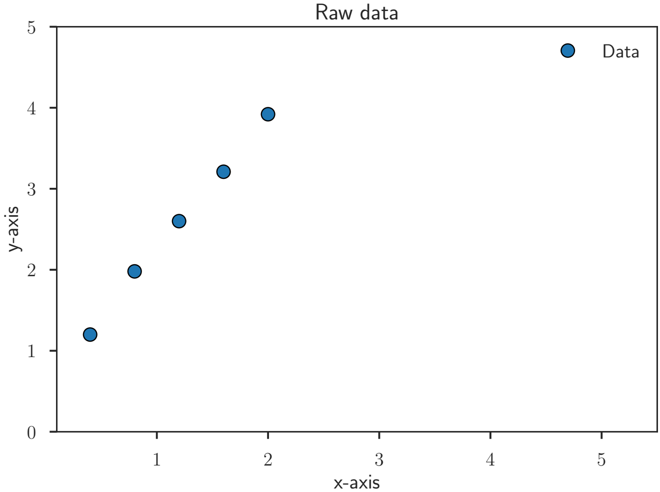
Ordinary Least Squares: toy example (y-corruption)



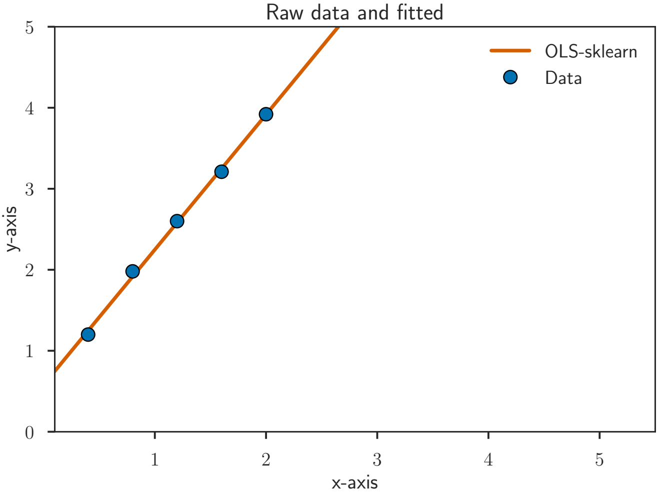
Ordinary Least Squares: toy example (y-corruption)



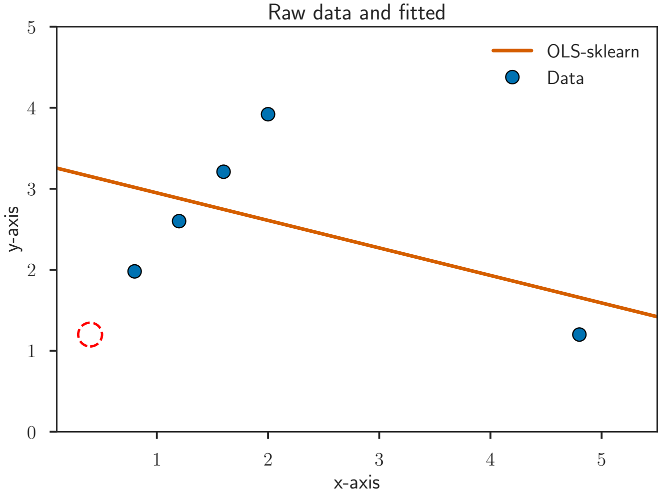
Ordinary Least Squares: toy example (x-corruption)



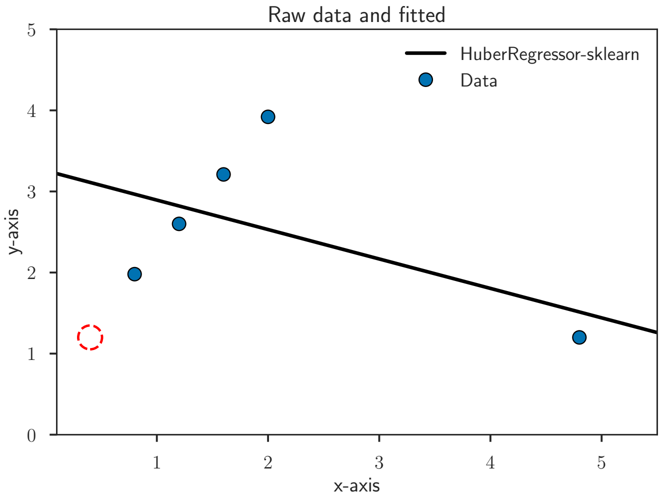
Ordinary Least Squares: toy example (x-corruption)



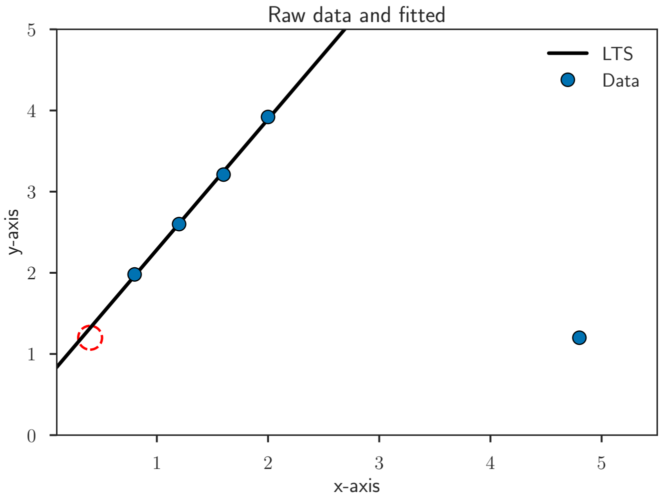
Ordinary Least Squares: toy example (x-corruption)



Ordinary Least Squares: toy example (x-corruption)

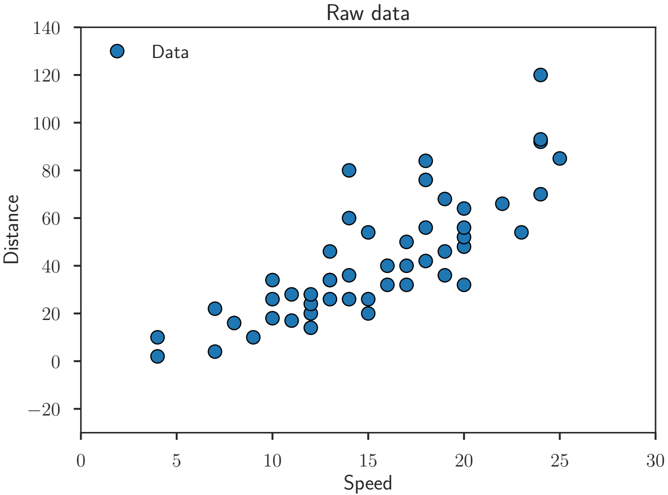


Ordinary Least Squares: toy example (x-corruption)



A real 2D example

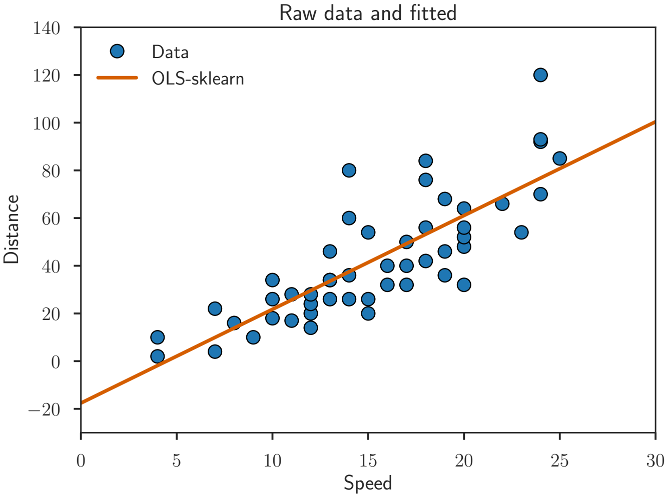
Example : braking distance for cars as a function of speed ($n = 50$ measurements)



Dataset *cars*: https://stat.ethz.ch/R-manual/R-devel/library/datasets/html/cars.html

A real 2D example

Example : braking distance for cars as a function of speed ($n = 50$ measurements)



Dataset *cars*: https://stat.ethz.ch/R-manual/R-devel/library/datasets/html/cars.html

Modeling : single feature

$$\text{Observations:} \quad (y_i, x_i), \text{ for } i = 1, \dots, n$$

Linear model or linear regression hypothesis assume:

$$\boxed{y_i \approx \beta_0^\star + \beta_1^\star x_i}$$

- $\beta_0^\star$: intercept (unknown)
- $\beta_1^\star$: slope (unknown)

Rem: both parameters are unknown from the statistician

Definitions

- y is an **observation** or a variable to explain
- x is a **feature** or a covariate

Modeling III

$$y_i = \beta_0^\star + \beta_1^\star x_i + \varepsilon_i$$

Definitions

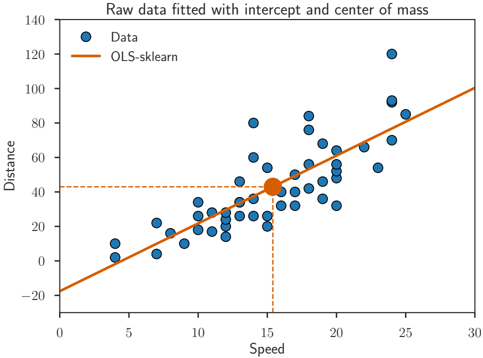
- **intercept** : the scalar $\beta_0^\star$
- **slope** : the scalar $\beta_1^\star$
- **noise** : the vector $\varepsilon = (\varepsilon_1, \ldots, \varepsilon_n)^\top$

Goal

Estimate $\beta_0^\star$ and $\beta_1^\star$ (unknown) by $\hat{\beta}_0$ and $\hat{\beta}_1$ relying on observations (y_i, x_i) for $i = 1, \ldots, n$

OLS and Center of gravity

$$y \approx \hat{\beta}_0 + \hat{\beta}_1 x$$



- $\overline{speed} = 15.4$
- $\overline{dist} = 42.98$
- $\hat{\beta}_0 = -17.579095$ intercept (negative!)
- $\hat{\beta}_1 = 3.932409$ slope

Physical interpretation: the cloud of points' center of gravity belongs to the (estimated) regression line

Centering

Centered model:

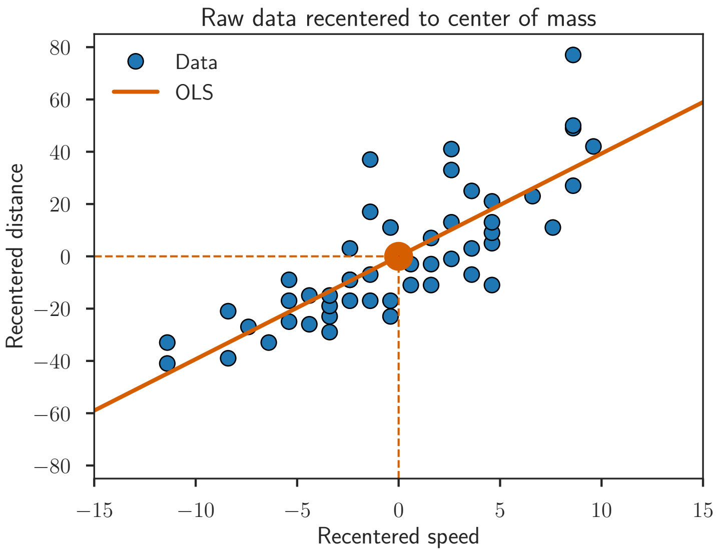
Write for any $i = 1, \ldots, n$: $\begin{cases} x'_i = x_i - \overline{x}_n \\ y'_i = y_i - \overline{y}_n \end{cases} \Leftrightarrow \begin{cases} \mathbf{x}' = \mathbf{x} - \overline{x}_n \mathbf{1}_n \\ \mathbf{y}' = \mathbf{y} - \overline{y}_n \mathbf{1}_n \end{cases}$

and $\mathbf{1}_n = (1, \ldots, 1)^\top \in \mathbb{R}^n$, then solving the OLS with $(\mathbf{x}', \mathbf{y}')$ leads to

$$\begin{cases} \widehat{\beta}'_0 = 0 \\ \widehat{\beta}'_1 = \dfrac{\dfrac{1}{n}\displaystyle\sum_{i=1}^n x'_i y'_i}{\dfrac{1}{n}\displaystyle\sum_{i=1}^n x_i'^2} \end{cases}$$

<u>Rem</u>: equivalent to choosing the cloud of points' center of mass as origin, *i.e.*, $(\overline{x}'_n, \overline{y}'_n) = (0, 0)$

Centering (II)



Centering and interpretation

Consider the coefficient $\widehat{\beta}_1'$ ($\widehat{\beta}_0' = 0$) for centered $\mathbf{y}', \mathbf{x}'$, then:

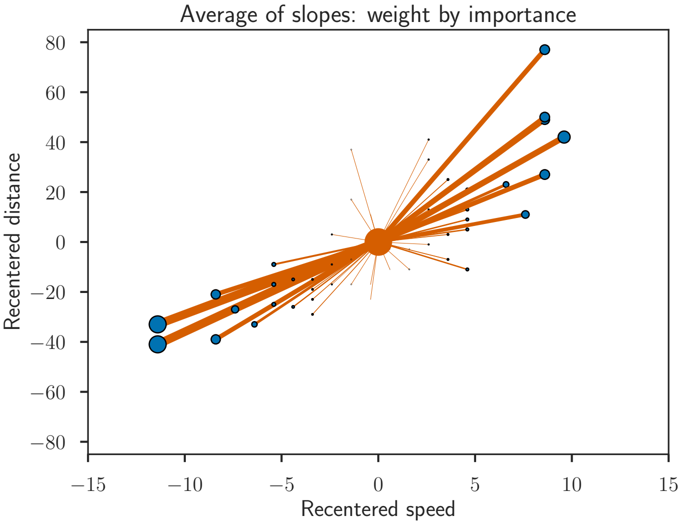
$$\widehat{\beta}_1' \in \underset{\beta_1 \in \mathbb{R}}{\arg\min} \sum_{i=1}^{n} (y_i' - \beta_1 x_i')^2 = \underset{\beta_1 \in \mathbb{R}}{\arg\min} \sum_{i=1}^{n} x_i'^2 \left(\frac{y_i'}{x_i'} - \beta_1 \right)^2$$

Interpretation: $\widehat{\beta}_1'$ is a weighted average of the slopes $\frac{y_i'}{x_i'}$

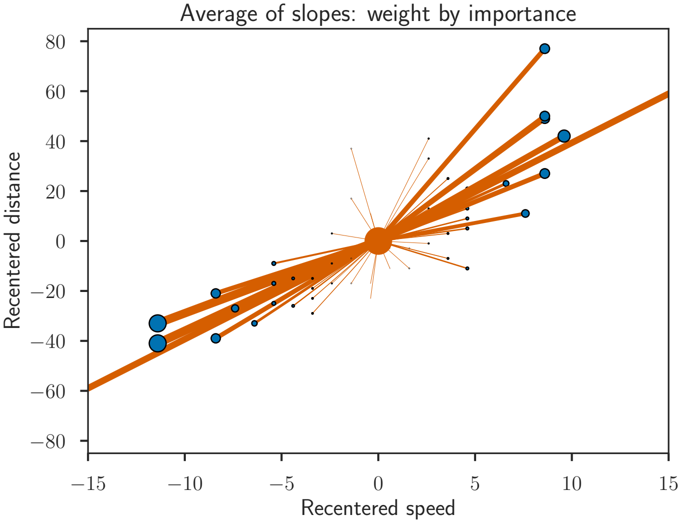
$$\widehat{\beta}_1' = \frac{\displaystyle\sum_{i=1}^{n} x_i'^2 \frac{y_i'}{x_i'}}{\displaystyle\sum_{j=1}^{n} x_j'^2}$$

Influence of extreme points: weights proportional to $x_i'^2$; **leverage** effect for far away points

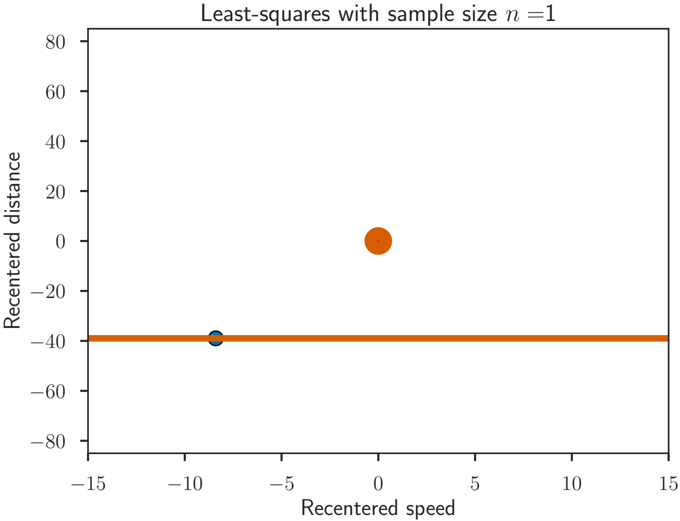
Extreme points – leverage effect



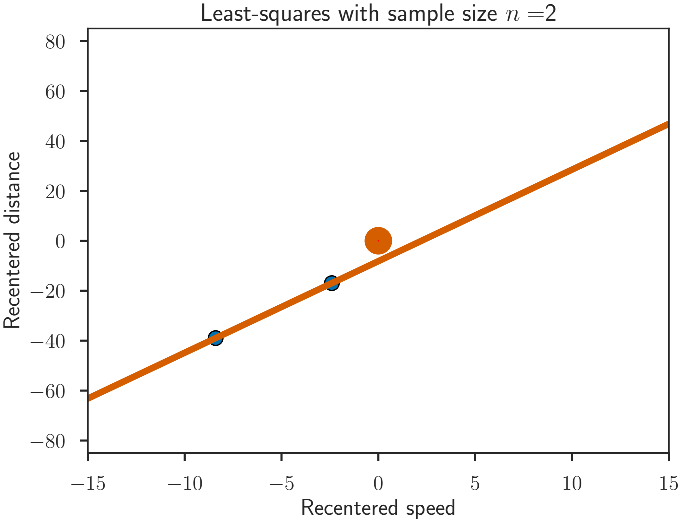
Extreme points – leverage effect



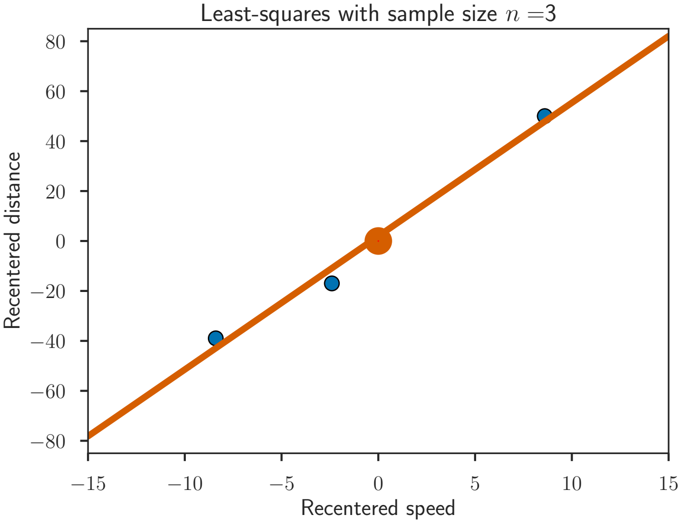
Extreme points – leverage effect



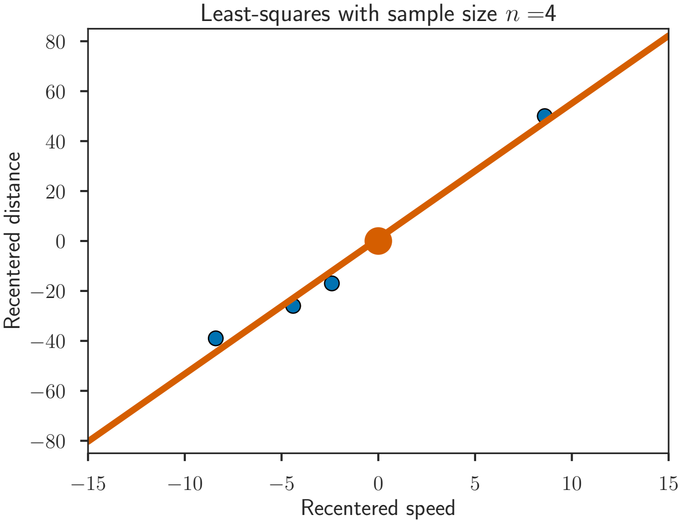
Extreme points – leverage effect



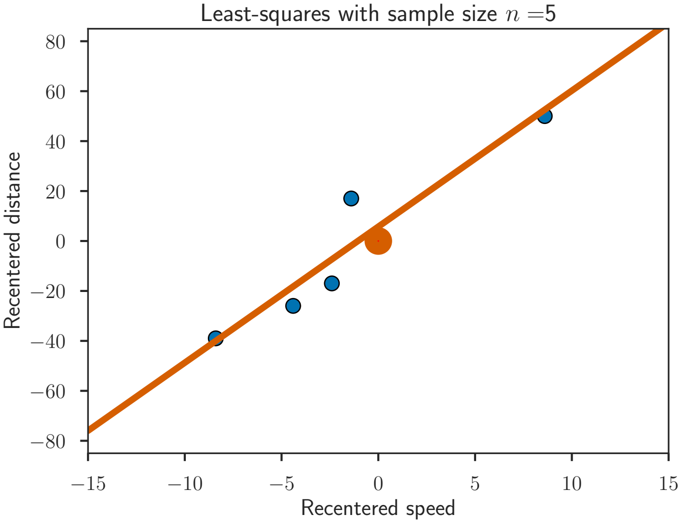
Extreme points – leverage effect



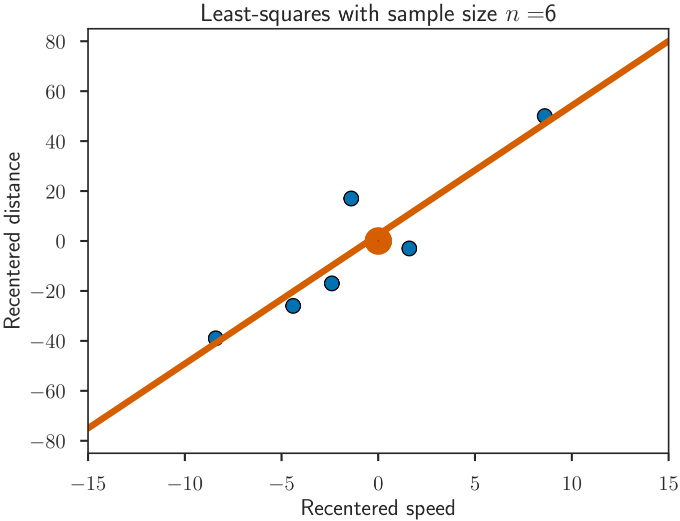
Extreme points – leverage effect



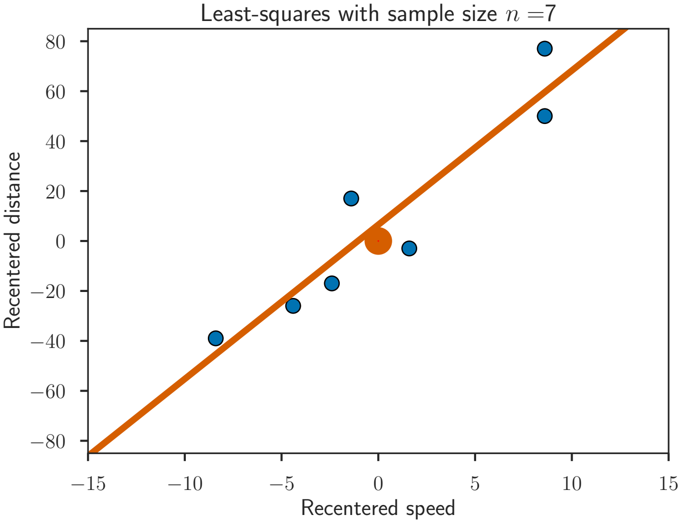
Extreme points – leverage effect



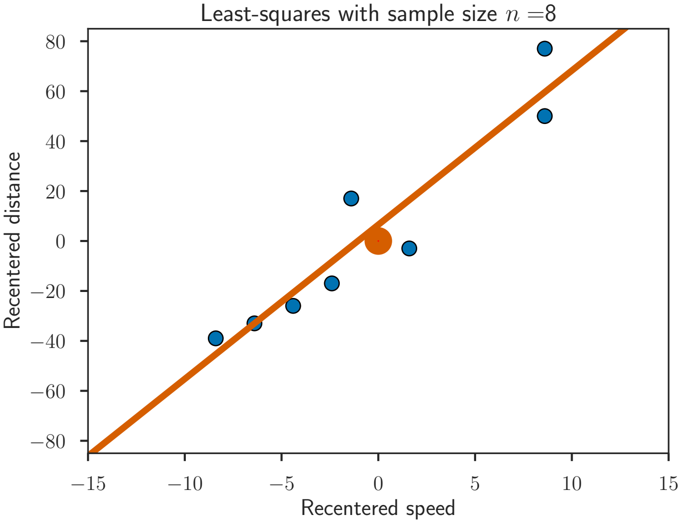
Extreme points – leverage effect



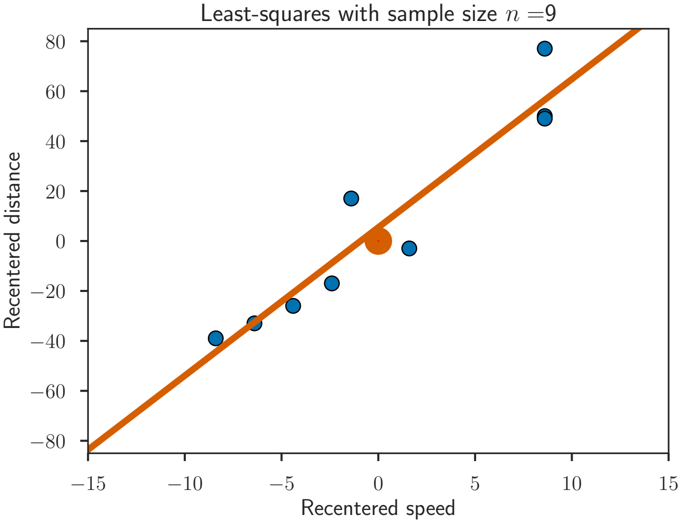
Extreme points – leverage effect



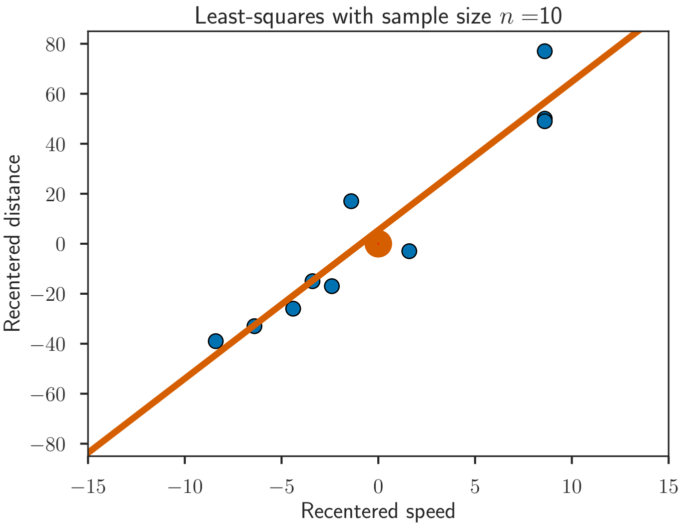
Extreme points – leverage effect



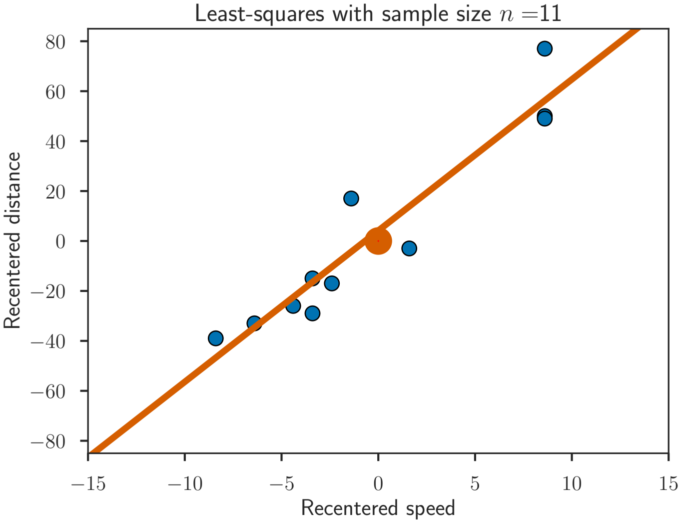
Extreme points – leverage effect



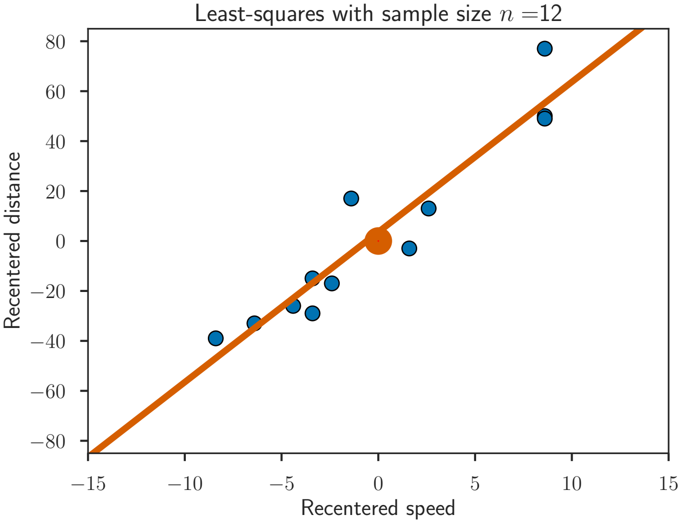
Extreme points – leverage effect



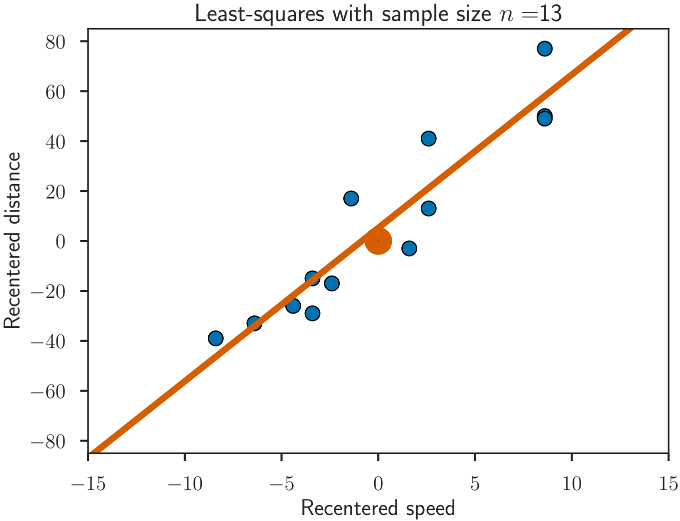
Extreme points – leverage effect



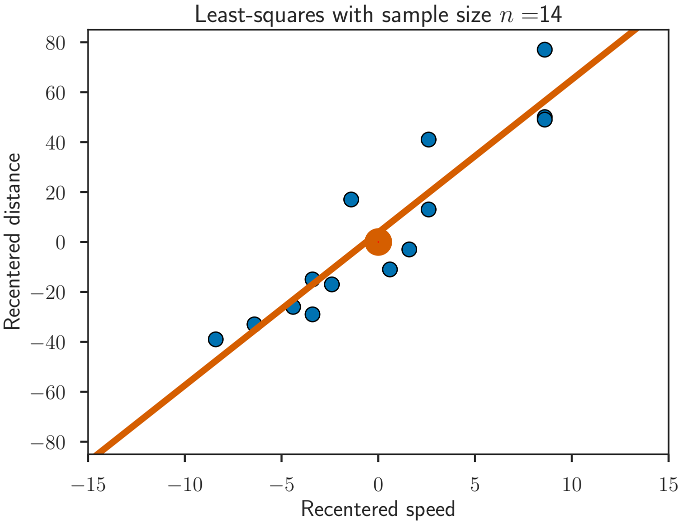
Extreme points – leverage effect



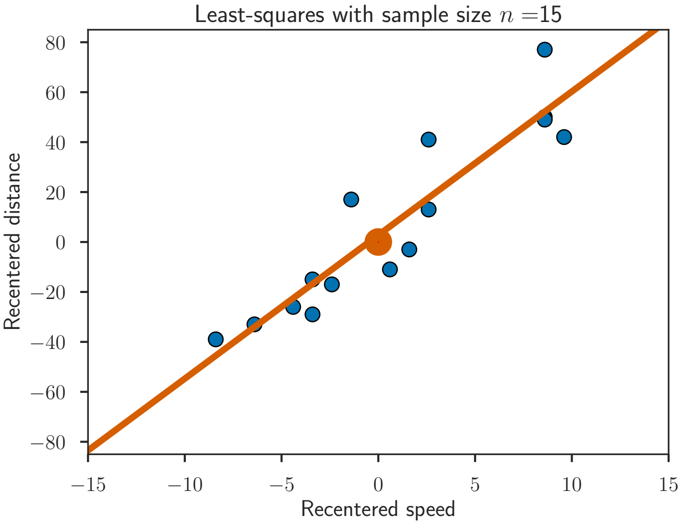
Extreme points – leverage effect



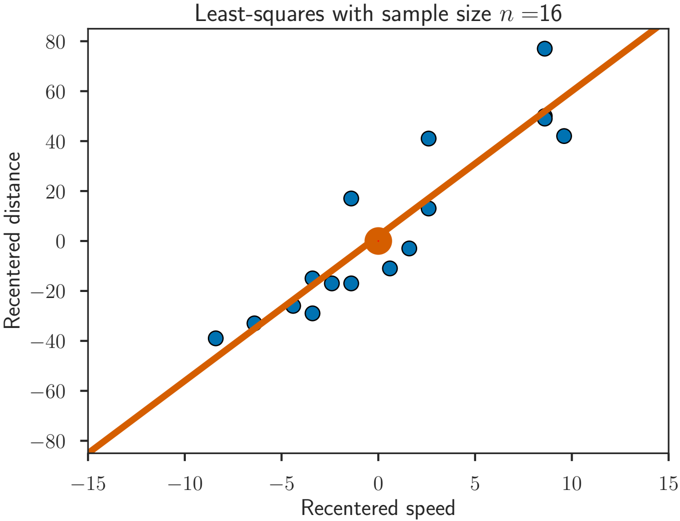
Extreme points – leverage effect



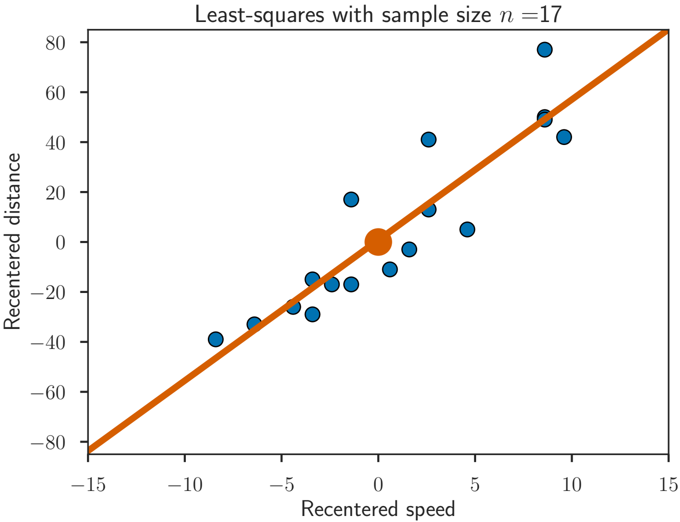
Extreme points – leverage effect



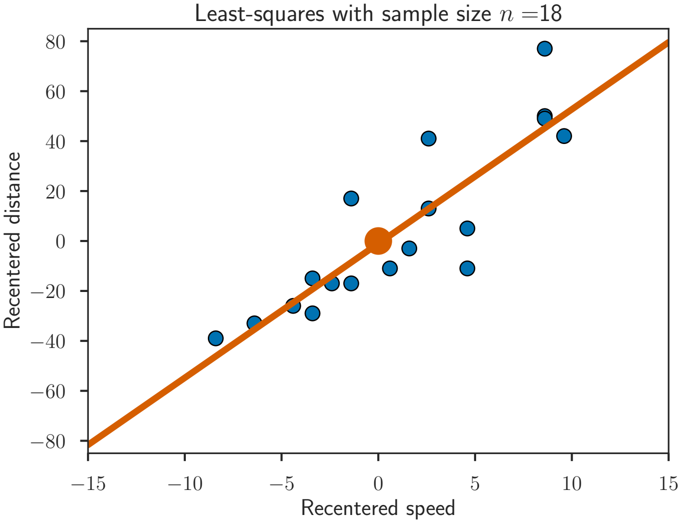
Extreme points – leverage effect



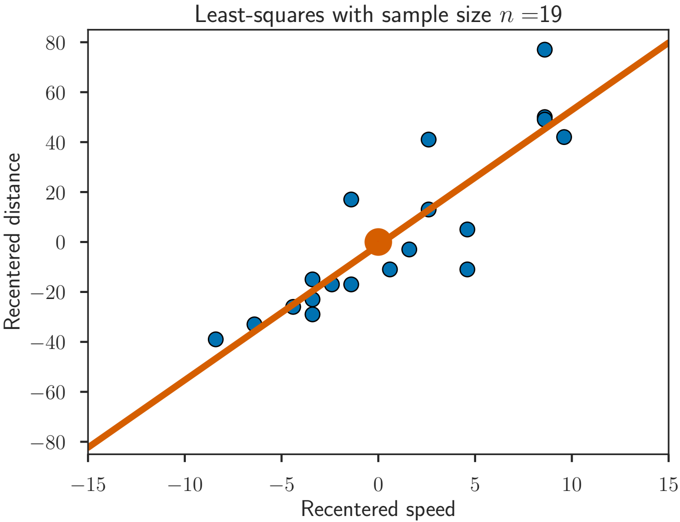
Extreme points – leverage effect



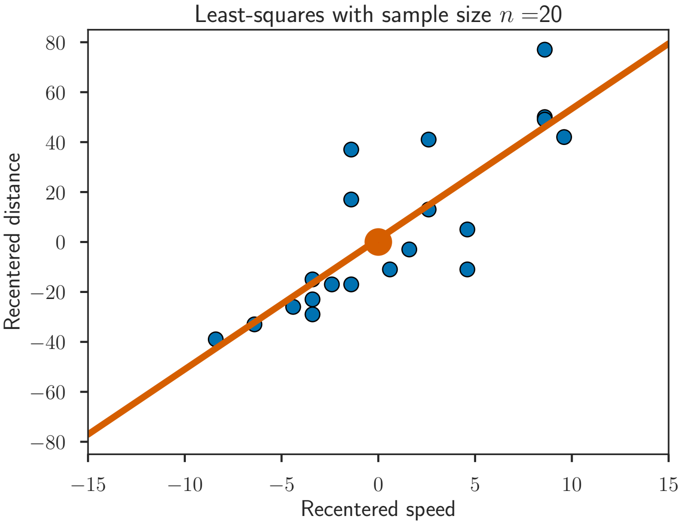
Extreme points – leverage effect



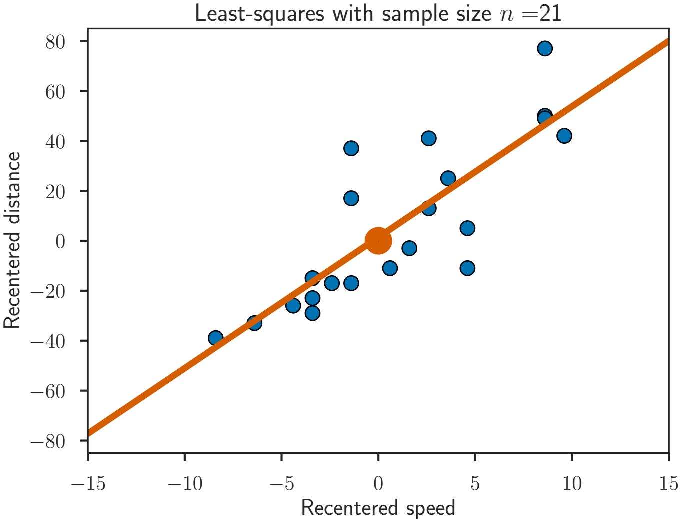
Extreme points – leverage effect



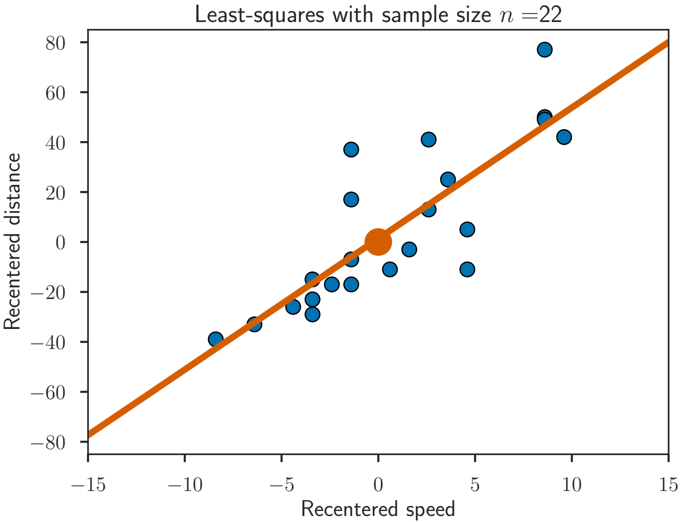
Extreme points – leverage effect



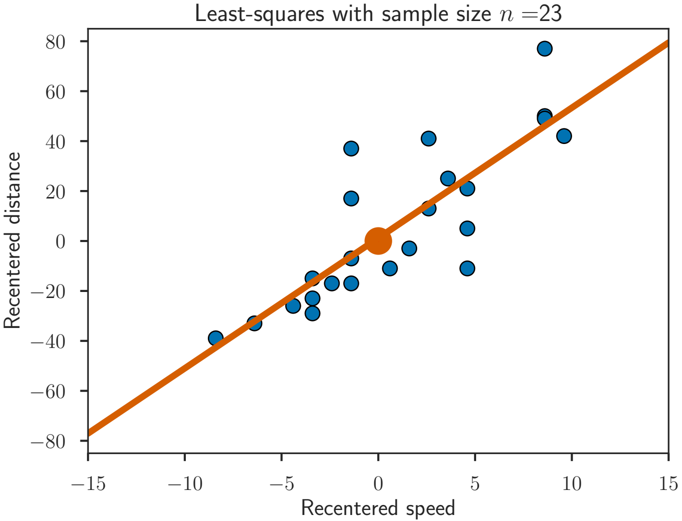
Extreme points – leverage effect



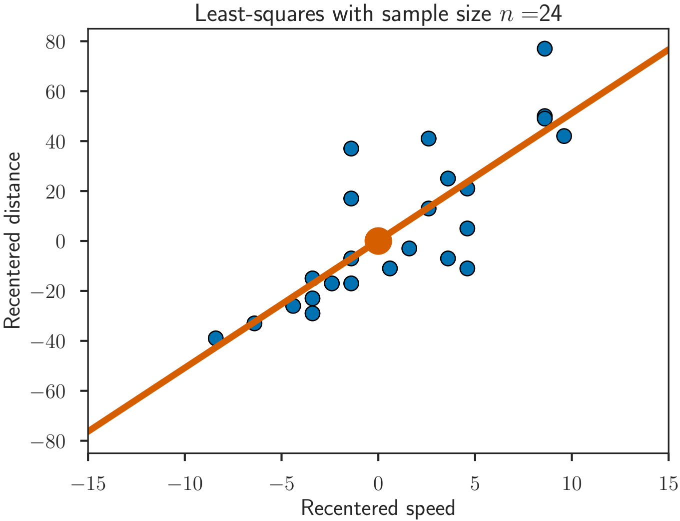
Extreme points – leverage effect



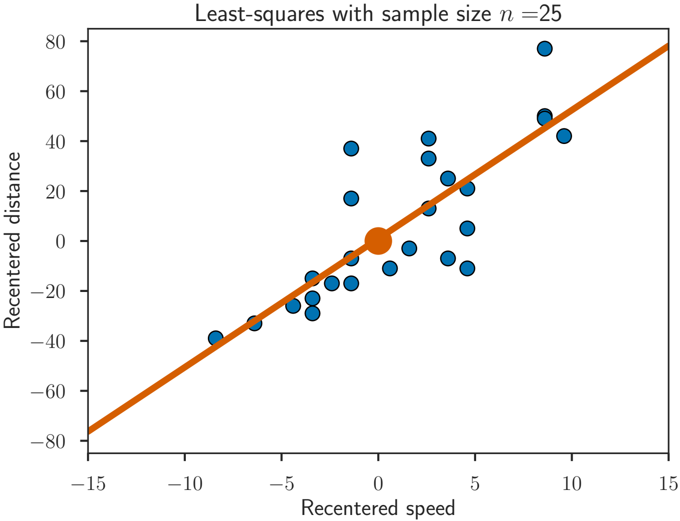
Extreme points – leverage effect



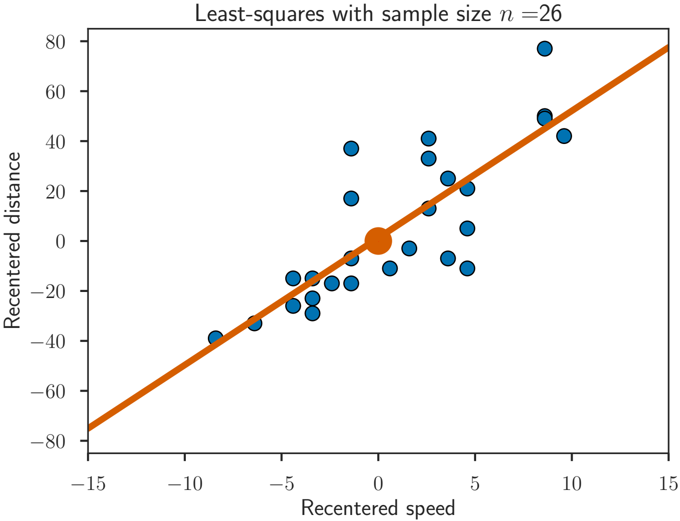
Extreme points – leverage effect



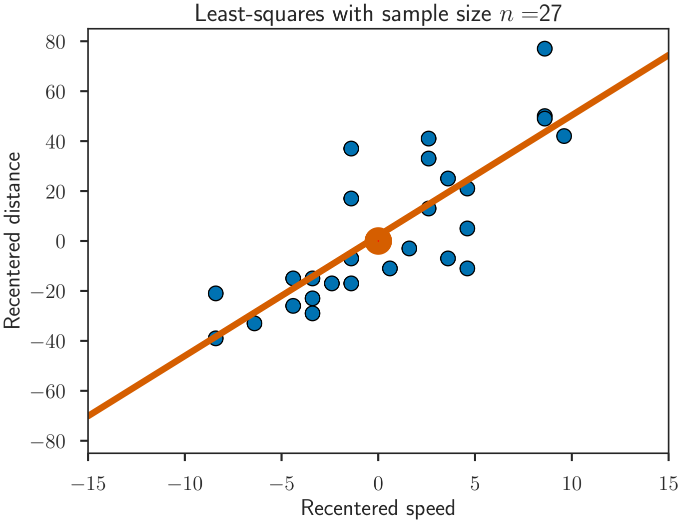
Extreme points – leverage effect



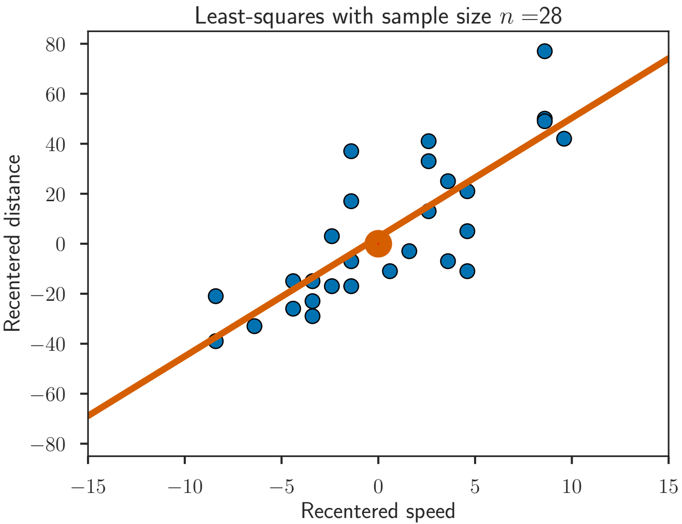
Extreme points – leverage effect



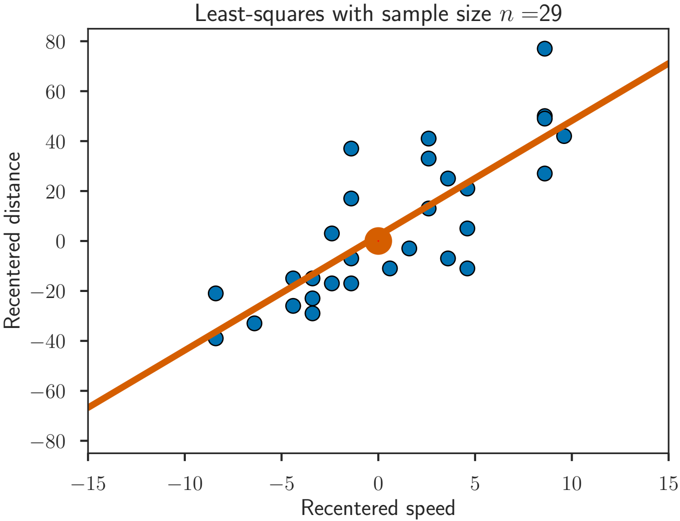
Extreme points – leverage effect



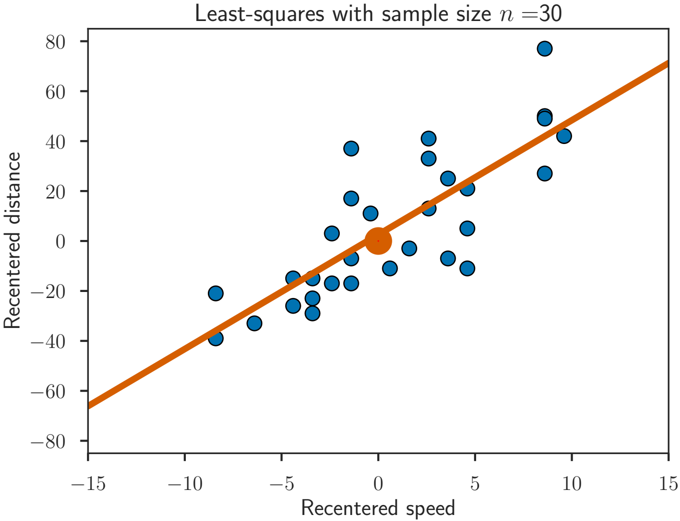
Extreme points – leverage effect



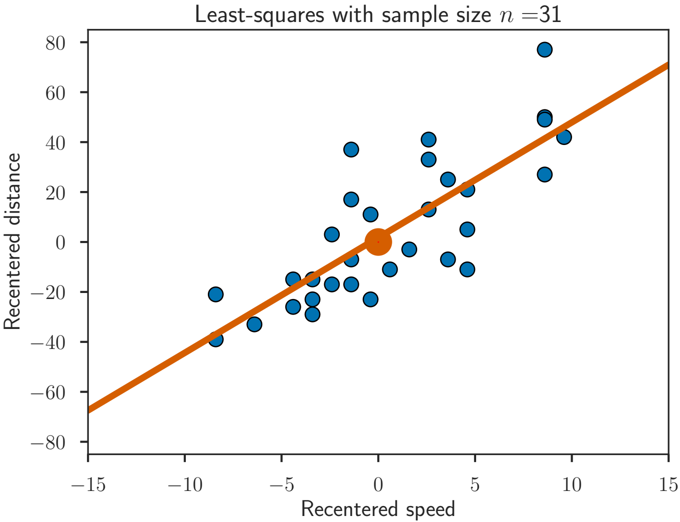
Extreme points – leverage effect



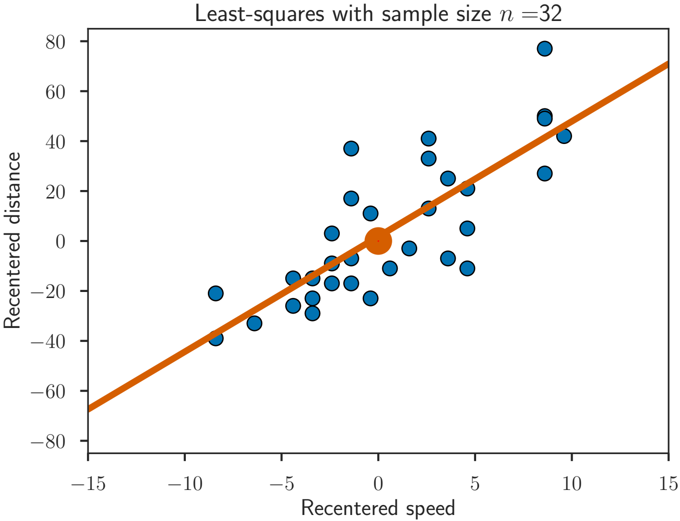
Extreme points – leverage effect



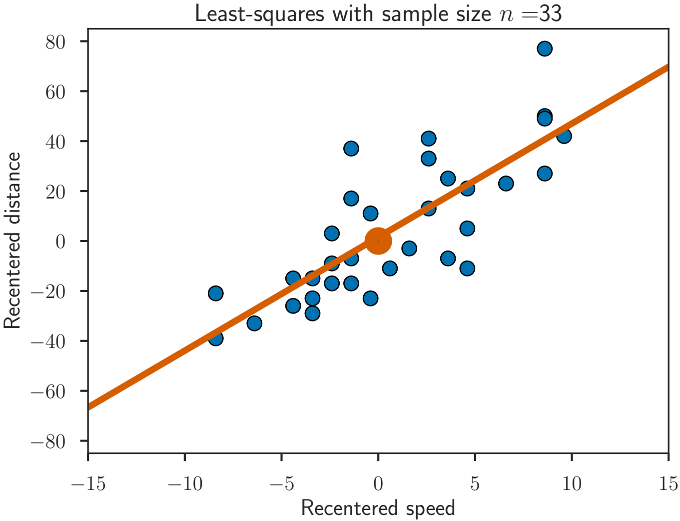
Extreme points – leverage effect



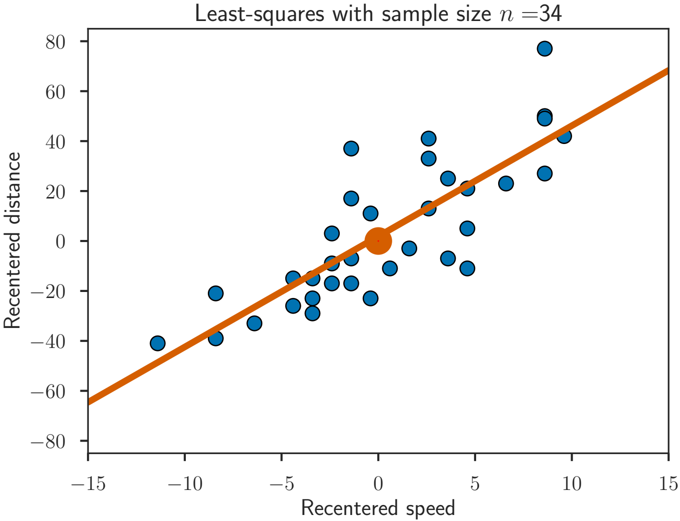
Extreme points – leverage effect



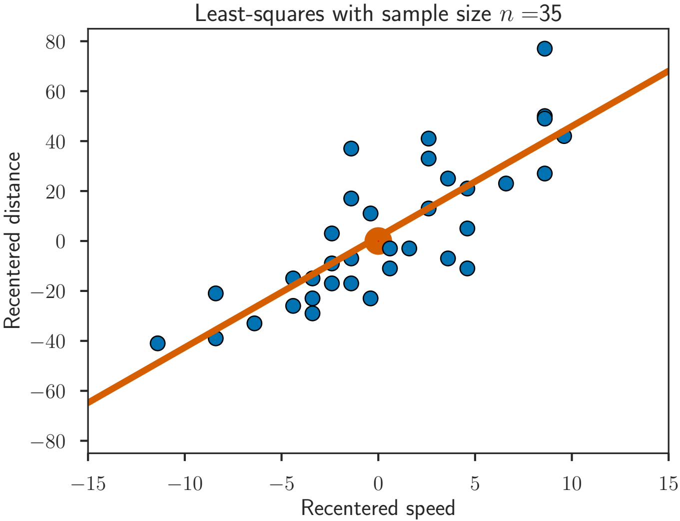
Extreme points – leverage effect



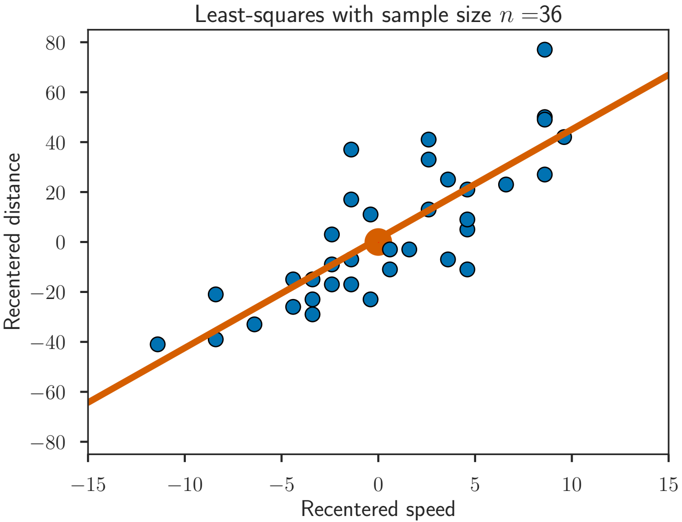
Extreme points – leverage effect



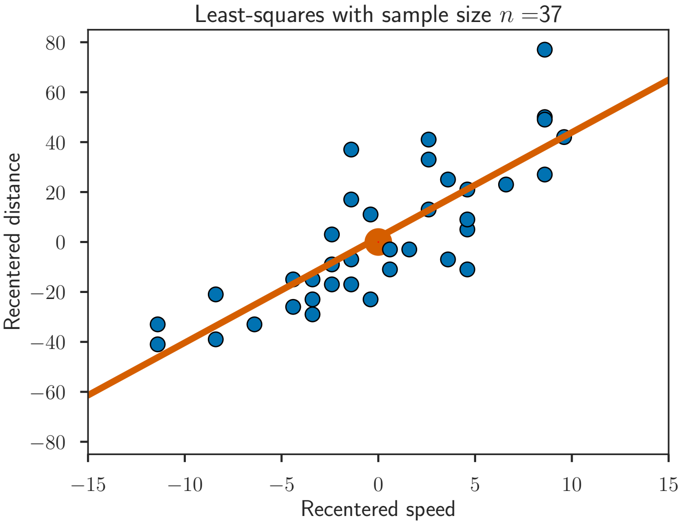
Extreme points – leverage effect



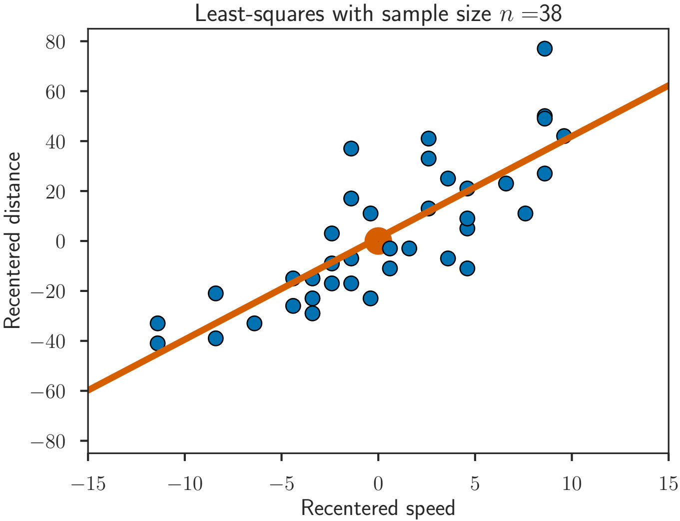
Extreme points – leverage effect



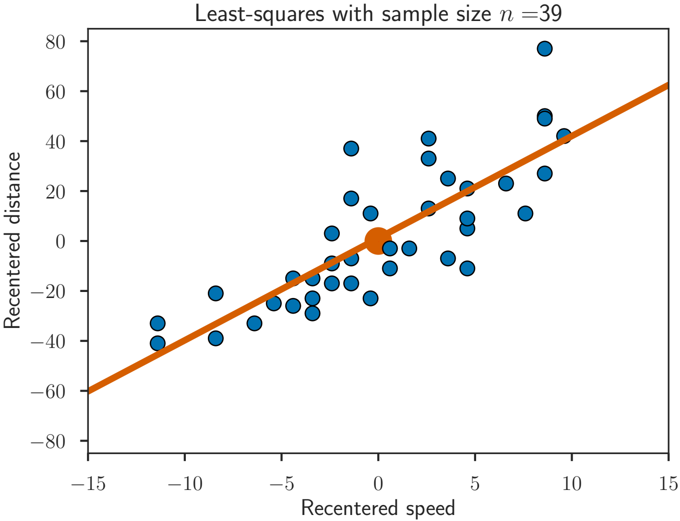
Extreme points – leverage effect



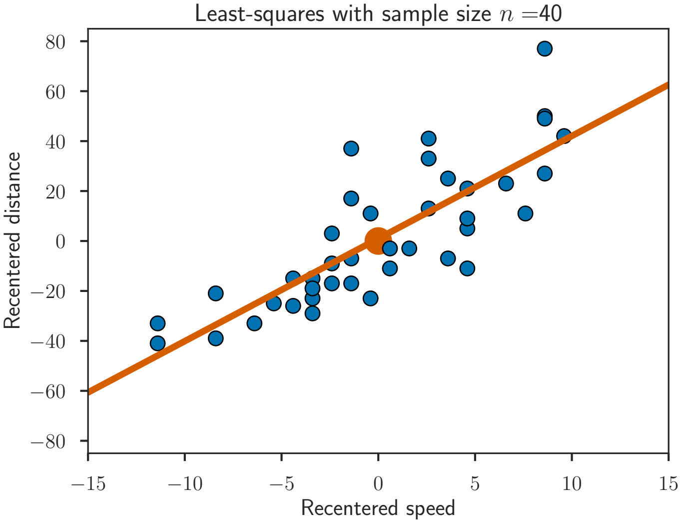
Extreme points – leverage effect



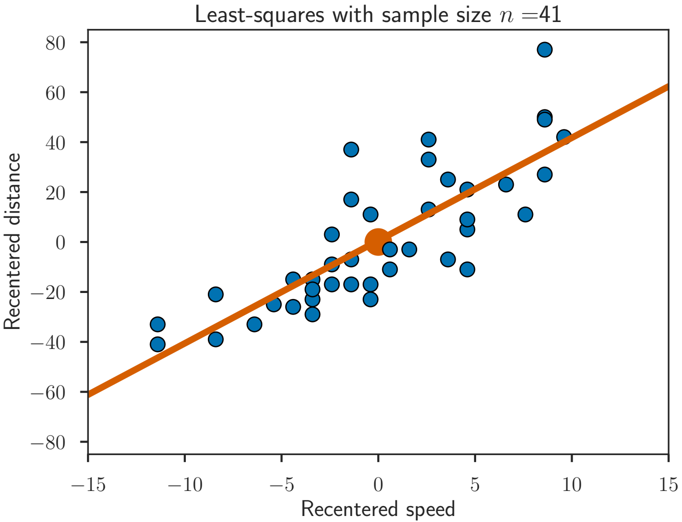
Extreme points – leverage effect



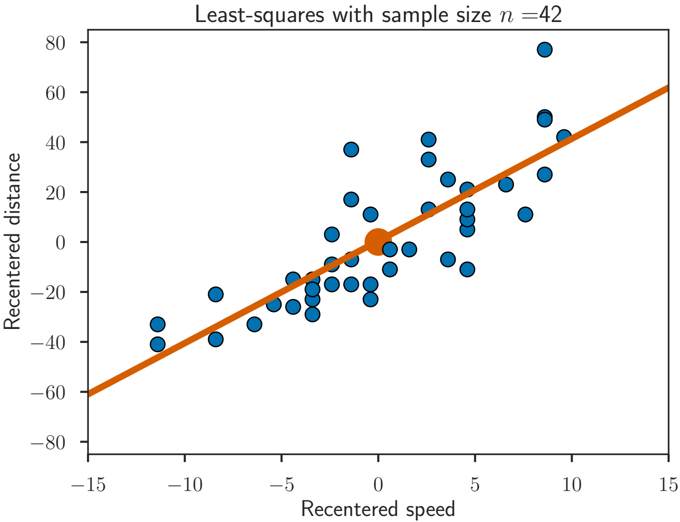
Extreme points – leverage effect



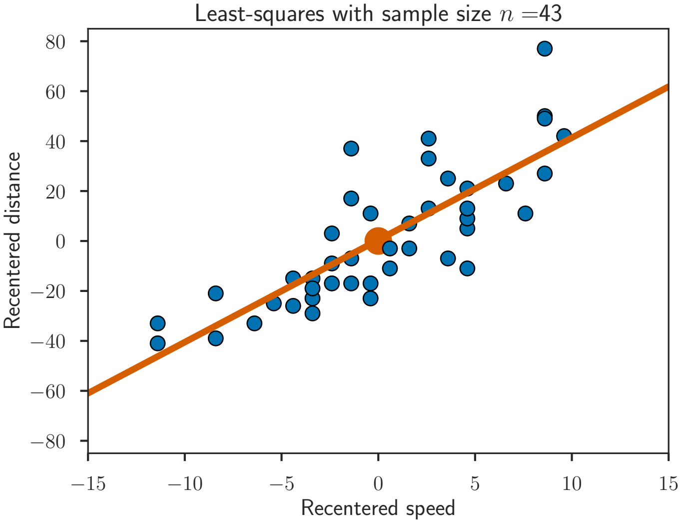
Extreme points – leverage effect



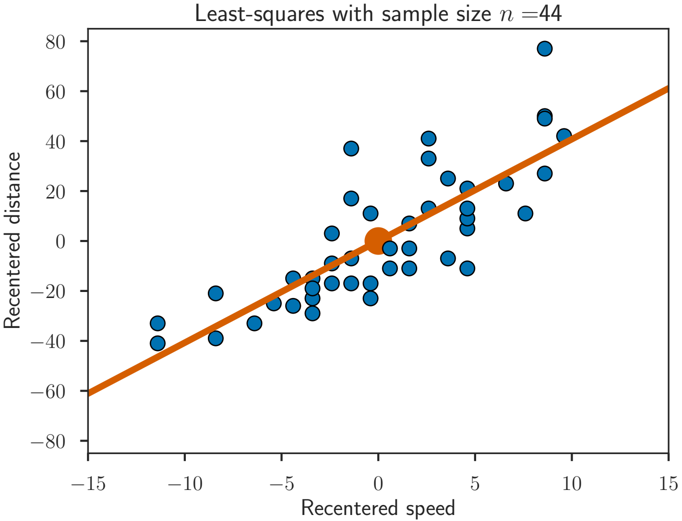
Extreme points – leverage effect



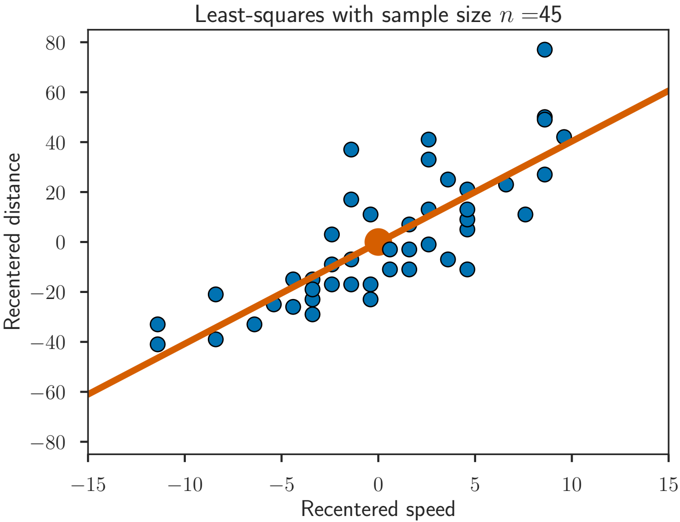
Extreme points – leverage effect



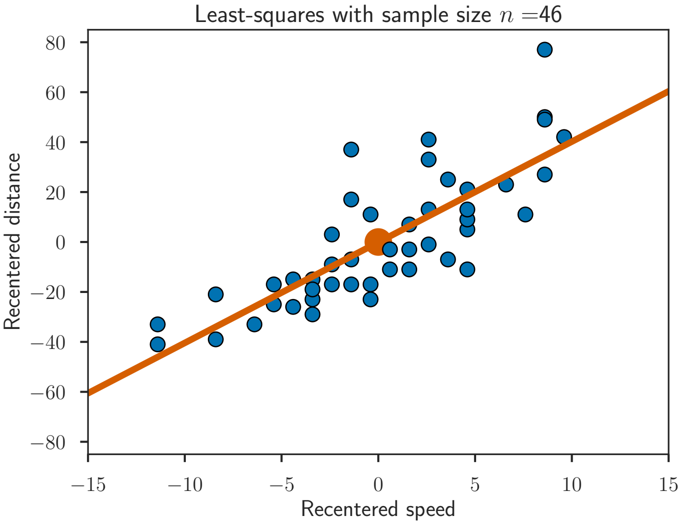
Extreme points – leverage effect



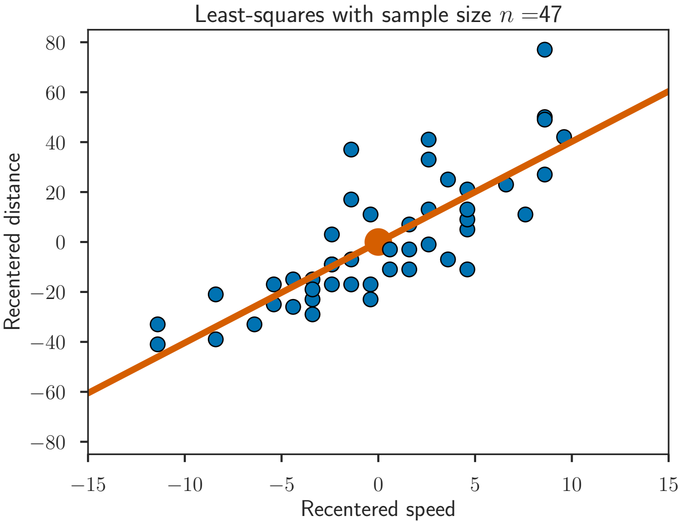
Extreme points – leverage effect



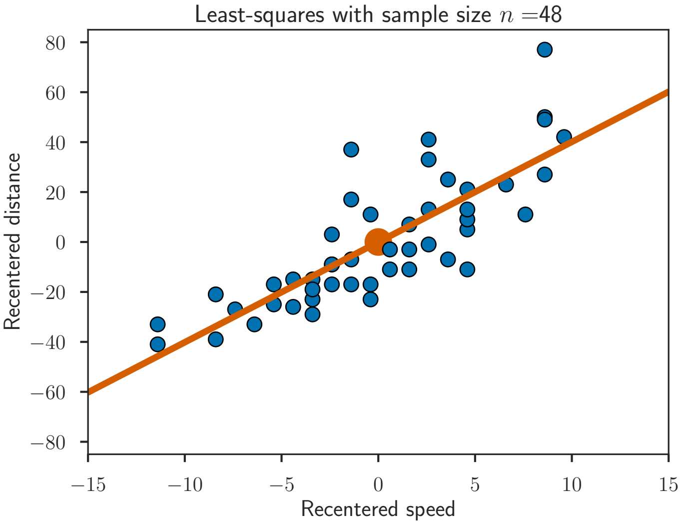
Extreme points – leverage effect



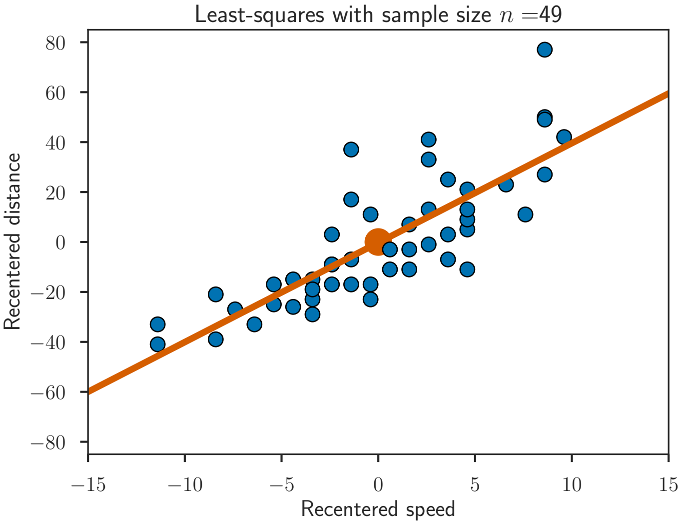
Extreme points – leverage effect



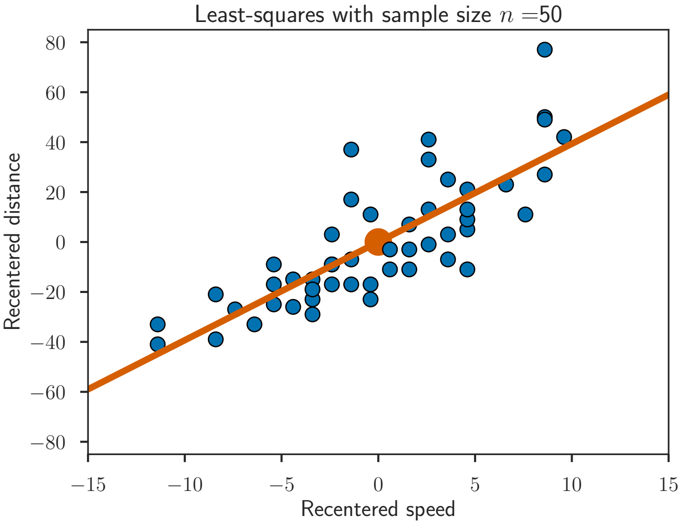
Extreme points – leverage effect



Extreme points – leverage effect



Extreme points – leverage effect



Multidimensional regression: Model / vocabulary

$$\boxed{\mathbf{y} = X\boldsymbol{\beta}^\star + \boldsymbol{\varepsilon}}$$

- $\mathbf{y} \in \mathbb{R}^n$: observations vector
- $X \in \mathbb{R}^{n \times p}$: **design** matrix (with features as columns)
- $\boldsymbol{\beta}^\star \in \mathbb{R}^p$: (unknown) **true** parameter to be estimated
- $\boldsymbol{\varepsilon} \in \mathbb{R}^n$: noise vector

<u>"Observations" point of view</u>: $y_i = \langle x_i, \boldsymbol{\beta}^\star \rangle + \varepsilon_i$ for $i = 1, \dots, n$
$\langle \cdot, \cdot \rangle$ stands for standard inner product

<u>"Features" point of view</u>: $\mathbf{y} = \sum_{j=1}^{p} \beta_j^\star \mathbf{x}_j + \boldsymbol{\varepsilon}$

(Ordinary) Least squares, (O)LS

A least square estimator is **any** solution of the following problem:

$$\hat{\boldsymbol{\beta}} \in \operatorname*{arg\,min}_{\boldsymbol{\beta} \in \mathbb{R}^p} \frac{1}{2} \|\mathbf{y} - X\boldsymbol{\beta}\|_2^2 := f(\beta)$$

$$\hat{\boldsymbol{\beta}} \in \operatorname*{arg\,min}_{\boldsymbol{\beta} \in \mathbb{R}^p} \frac{1}{2} \sum_{i=1}^{n} \left[y_i - \langle x_i, \boldsymbol{\beta} \rangle\right]^2$$

Rem: uniqueness does not hold when features are **co-linear**, and then there are an infinite number of solutions

Rem: an intercept is often added

Rem: Gaussian (-log)-likelihood leads to square formulation

Least squares - normal equation

$$\nabla f(\boldsymbol{\beta}) = 0 \Leftrightarrow X^\top X \boldsymbol{\beta} - X^\top \mathbf{y} = X^\top (X\boldsymbol{\beta} - \mathbf{y}) = 0$$

Theorem

Fermat's rule ensures that any LS solution $\hat{\beta}$ satisfies:

Normal equation: $$\boxed{X^\top X \hat{\boldsymbol{\beta}} = X^\top \mathbf{y}}$$

$\hat{\beta}$ is solution of the linear system "$A\boldsymbol{\beta} = b$" for a matrix $A = X^\top X$ and right hand side $b = X^\top \mathbf{y}$

Proof: gradient computation

The gradient of f, ∇f is defined for any $\boldsymbol{\beta}$ as the vector satisfying:

$$f(\boldsymbol{\beta}+h) = f(\boldsymbol{\beta}) + \langle h, \nabla f(\boldsymbol{\beta})\rangle + o(h) \quad \text{for any } h$$

For the f of interest here, this reads

$$f(\boldsymbol{\beta}+h) = \frac{1}{2}\|\mathbf{y}\|^2 - \langle \boldsymbol{\beta}+h, X^\top \mathbf{y}\rangle + \frac{1}{2}(\boldsymbol{\beta}+h)^\top X^\top X(\boldsymbol{\beta}+h)$$

Proof: gradient computation

The gradient of f, ∇f is defined for any $\boldsymbol{\beta}$ as the vector satisfying:

$$f(\boldsymbol{\beta} + h) = f(\boldsymbol{\beta}) + \langle h, \nabla f(\boldsymbol{\beta}) \rangle + o(h) \quad \text{for any } h$$

For the f of interest here, this reads

$$\begin{aligned} f(\boldsymbol{\beta} + h) =& \frac{1}{2}\|\mathbf{y}\|^2 - \langle \boldsymbol{\beta} + h, X^\top \mathbf{y} \rangle + \frac{1}{2}(\boldsymbol{\beta} + h)^\top X^\top X (\boldsymbol{\beta} + h) \\ =& \frac{1}{2}\|\mathbf{y}\|^2 - \langle \boldsymbol{\beta}, X^\top \mathbf{y} \rangle - \langle h, X^\top \mathbf{y} \rangle \\ & + \frac{1}{2}\boldsymbol{\beta}^\top X^\top X \boldsymbol{\beta} + \frac{1}{2} h^\top X^\top X h + \boldsymbol{\beta}^\top X^\top X h \end{aligned}$$

Proof: gradient computation

The gradient of f, ∇f is defined for any $\boldsymbol{\beta}$ as the vector satisfying:

$$f(\boldsymbol{\beta}+h) = f(\boldsymbol{\beta}) + \langle h, \nabla f(\boldsymbol{\beta})\rangle + o(h) \quad \text{for any } h$$

For the f of interest here, this reads

$$\begin{aligned} f(\boldsymbol{\beta}+h) =& \frac{1}{2}\|\mathbf{y}\|^2 - \langle \boldsymbol{\beta}+h, X^\top \mathbf{y}\rangle + \frac{1}{2}(\boldsymbol{\beta}+h)^\top X^\top X(\boldsymbol{\beta}+h) \\ =& \frac{1}{2}\|\mathbf{y}\|^2 - \langle \boldsymbol{\beta}, X^\top \mathbf{y}\rangle - \langle h, X^\top \mathbf{y}\rangle \\ & + \frac{1}{2}\boldsymbol{\beta}^\top X^\top X\boldsymbol{\beta} + \frac{1}{2}h^\top X^\top X h + \boldsymbol{\beta}^\top X^\top X h \\ =& f(\boldsymbol{\beta}) - \langle h, X^\top \mathbf{y}\rangle + \frac{1}{2}h^\top X^\top X h + \boldsymbol{\beta}^\top X^\top X h \end{aligned}$$

Proof: gradient computation

The gradient of f, ∇f is defined for any $\boldsymbol{\beta}$ as the vector satisfying:

$$f(\boldsymbol{\beta}+h) = f(\boldsymbol{\beta}) + \langle h, \nabla f(\boldsymbol{\beta})\rangle + o(h) \qquad \text{for any } h$$

For the f of interest here, this reads

$$\begin{aligned}
f(\boldsymbol{\beta}+h) =& \frac{1}{2}\|\mathbf{y}\|^2 - \langle \boldsymbol{\beta}+h, X^\top \mathbf{y}\rangle + \frac{1}{2}(\boldsymbol{\beta}+h)^\top X^\top X(\boldsymbol{\beta}+h) \\
=& \frac{1}{2}\|\mathbf{y}\|^2 - \langle \boldsymbol{\beta}, X^\top \mathbf{y}\rangle - \langle h, X^\top \mathbf{y}\rangle \\
&+ \frac{1}{2}\boldsymbol{\beta}^\top X^\top X\boldsymbol{\beta} + \frac{1}{2}h^\top X^\top X h + \boldsymbol{\beta}^\top X^\top X h \\
=& f(\boldsymbol{\beta}) - \langle h, X^\top \mathbf{y}\rangle + \frac{1}{2}h^\top X^\top X h + \boldsymbol{\beta}^\top X^\top X h \\
=& f(\boldsymbol{\beta}) + \langle h, \underbrace{X^\top X\boldsymbol{\beta} - X^\top \mathbf{y}}_{\nabla f(\boldsymbol{\beta})}\rangle + \underbrace{\frac{1}{2}h^\top X^\top X h}_{o(h)}
\end{aligned}$$

Proof: gradient computation

The gradient of f, ∇f is defined for any $\boldsymbol{\beta}$ as the vector satisfying:

$$f(\boldsymbol{\beta} + h) = f(\boldsymbol{\beta}) + \langle h, \nabla f(\boldsymbol{\beta}) \rangle + o(h) \quad \text{for any } h$$

For the f of interest here, this reads

$$\begin{aligned}
f(\boldsymbol{\beta} + h) =& \frac{1}{2}\|\mathbf{y}\|^2 - \langle \boldsymbol{\beta} + h, X^\top \mathbf{y} \rangle + \frac{1}{2}(\boldsymbol{\beta} + h)^\top X^\top X(\boldsymbol{\beta} + h) \\
=& \frac{1}{2}\|\mathbf{y}\|^2 - \langle \boldsymbol{\beta}, X^\top \mathbf{y} \rangle - \langle h, X^\top \mathbf{y} \rangle \\
& + \frac{1}{2}\boldsymbol{\beta}^\top X^\top X \boldsymbol{\beta} + \frac{1}{2} h^\top X^\top X h + \boldsymbol{\beta}^\top X^\top X h \\
=& f(\boldsymbol{\beta}) - \langle h, X^\top \mathbf{y} \rangle + \frac{1}{2} h^\top X^\top X h + \boldsymbol{\beta}^\top X^\top X h \\
=& f(\boldsymbol{\beta}) + \langle h, \underbrace{X^\top X \boldsymbol{\beta} - X^\top \mathbf{y}}_{\nabla f(\boldsymbol{\beta})} \rangle + \underbrace{\frac{1}{2} h^\top X^\top X h}_{o(h)}
\end{aligned}$$

Hence, $\boxed{\nabla f(\boldsymbol{\beta}) = X^\top X \boldsymbol{\beta} - X^\top \mathbf{y} = X^\top (X\boldsymbol{\beta} - \mathbf{y})}$

Proof: gradient computation

The gradient of f, ∇f is defined for any $\boldsymbol{\beta}$ as the vector satisfying:

$$f(\boldsymbol{\beta}+h) = f(\boldsymbol{\beta}) + \langle h, \nabla f(\boldsymbol{\beta})\rangle + o(h) \qquad \text{for any } h$$

For the f of interest here, this reads

$$\begin{aligned}
f(\boldsymbol{\beta}+h) =& \frac{1}{2}\|\mathbf{y}\|^2 - \langle \boldsymbol{\beta}+h, X^\top \mathbf{y}\rangle + \frac{1}{2}(\boldsymbol{\beta}+h)^\top X^\top X(\boldsymbol{\beta}+h) \\
=& \frac{1}{2}\|\mathbf{y}\|^2 - \langle \boldsymbol{\beta}, X^\top \mathbf{y}\rangle - \langle h, X^\top \mathbf{y}\rangle \\
& + \frac{1}{2}\boldsymbol{\beta}^\top X^\top X\boldsymbol{\beta} + \frac{1}{2}h^\top X^\top X h + \boldsymbol{\beta}^\top X^\top X h \\
=& f(\boldsymbol{\beta}) - \langle h, X^\top \mathbf{y}\rangle + \frac{1}{2}h^\top X^\top X h + \boldsymbol{\beta}^\top X^\top X h \\
=& f(\boldsymbol{\beta}) + \langle h, \underbrace{X^\top X\boldsymbol{\beta} - X^\top \mathbf{y}}_{\nabla f(\boldsymbol{\beta})}\rangle + \underbrace{\frac{1}{2}h^\top X^\top X h}_{o(h)}
\end{aligned}$$

Hence, $$\boxed{\nabla f(\boldsymbol{\beta}) = X^\top X\boldsymbol{\beta} - X^\top \mathbf{y} = X^\top(X\boldsymbol{\beta} - \mathbf{y})}$$

Vocabulary (and abuse of terms)

Definition

We call **Gramian matrix** the matrix $X^\top X \in \mathbb{R}^{p\times p}$, whose general term is $[X^\top X]_{i,j} = \langle \mathbf{x}_i, \mathbf{x}_j \rangle$

Rem: $X^\top X$ is often referred to as the feature correlation matrix (true for standardized columns)

Rem: when columns are scaled such that $\forall j \in [\![1, p]\!], \|\mathbf{x}_j\|^2 = n$, the Gramian diagonal is $(n, \ldots, n)$

$$X^\top \mathbf{y} = \begin{pmatrix} \langle \mathbf{x}_1, \mathbf{y} \rangle \\ \vdots \\ \langle \mathbf{x}_p, \mathbf{y} \rangle \end{pmatrix}$$: observations/features correlation

OLS closed-form solution (full rank case)

Theorem

If X is full (column) rank (*i.e.*, if $X^\top X$ is non-singular) then

$$\hat{\boldsymbol{\beta}}^{\mathrm{OLS}} = (X^\top X)^{-1} X^\top \mathbf{y}$$

Rem: if $X = \mathbf{1}_n$: $\hat{\boldsymbol{\beta}}^{\mathrm{OLS}} = \dfrac{\langle \mathbf{1}_n, \mathbf{y} \rangle}{\langle \mathbf{1}_n, \mathbf{1}_n \rangle} = \bar{y}_n$ (empirical mean)

Rem: single feature $X = \mathbf{x} = (x_1, \ldots, x_n)^\top$: $\hat{\boldsymbol{\beta}}^{\mathrm{OLS}} = \langle \dfrac{\mathbf{x}}{\|\mathbf{x}\|^2}, \mathbf{y} \rangle$

Beware: in practice **avoid** inverting the matrix $X^\top X$

- numerically time consuming
- the matrix $X^\top X$ is not even be invertible if "$p \gg n$", *e.g.*, in biology n patients (≈ 100), p genes (≈ 50000)

Example

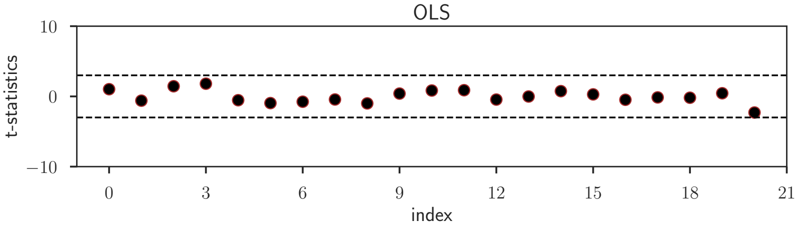
Stackloss dataset:
"Stackloss plant data, Brownlee (1965), contains 21 days of measurements from a plant's oxidation of ammonia to nitric acid. The nitric oxide pollutants are captured in an absorption tower."

- number of samples : $n = 21$
- number of features : $p = 3$
- $\mathbf{y}$ (to predict): STACKLOSS - 10 times the percentage of ammonia going into the plant escaping from the tower

Features:

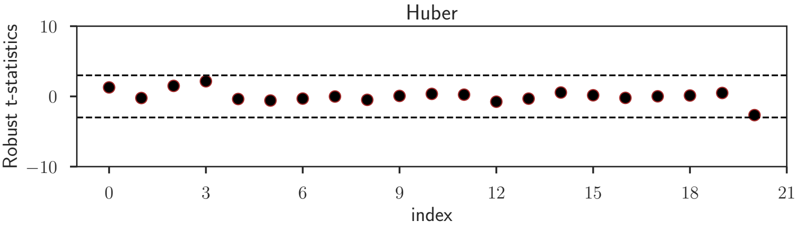
- AIRFLOW - Rate of operation of the plant
- WATERTEMP - Cooling water temperature in the tower
- ACIDCONC - Acid concentration of circulating acid minus 50 times 10.

3σ rule to spot outliers in a linear model



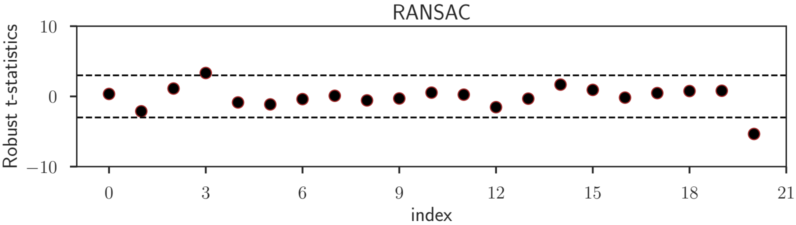
$$t_i = \frac{y_i - \langle x_i, \hat{\beta}^{\mathrm{OLS}} \rangle}{\hat{\sigma}} \text{ with } \hat{\sigma} = \frac{\left\| y - X\hat{\beta}^{\mathrm{OLS}} \right\|}{n-p}$$

3σ rule to spot outliers in a linear model



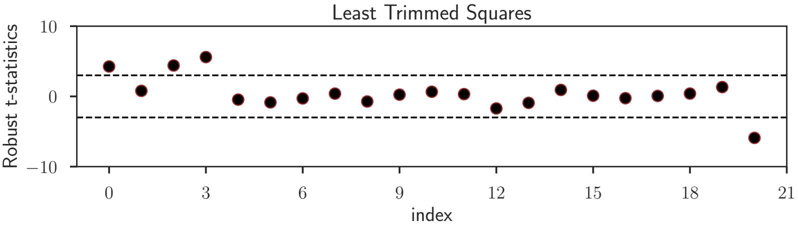
$$t_i = \frac{y_i - \langle x_i, \hat{\beta}^{\text{Huber}} \rangle}{\hat{\sigma}} \text{ with } \hat{\sigma} = \frac{\left\| y - X\hat{\beta}^{\text{Huber}} \right\|}{n-p}$$

3σ rule to spot outliers in a linear model



$$t_i = \frac{y_i - \langle x_i, \hat{\beta}^{\mathrm{RANSAC}} \rangle}{\hat{\sigma}} \text{ with } \hat{\sigma} = \frac{\mathrm{MAD}_n(y - X\hat{\beta}^{\mathrm{RANSAC}})}{0.6745}$$

3σ rule to spot outliers in a linear model



$$t_i = \frac{y_i - \langle x_i, \hat{\beta}^{\mathrm{LTS}} \rangle}{\hat{\sigma}} \text{ with } \hat{\sigma} = \frac{\mathrm{MAD}_n(y - X\hat{\beta}^{\mathrm{LTS}})}{0.6745}$$

References I

- Alfons, A., C. Croux, and S. Gelper. "Sparse least trimmed squares regression for analyzing high-dimensional large data sets". In: *Ann. Appl. Stat.* 7.1 (2013), pp. 226–248.
- Avella-Medina, M. and E. M. Ronchetti. "Robust and consistent variable selection in high-dimensional generalized linear models". In: *Biometrika* 105.1 (2018), pp. 31–44.
- Bauschke, H. H. and P. L. Combettes. *Convex analysis and monotone operator theory in Hilbert spaces*. New York: Springer, 2011, pp. xvi+468.
- Beck, A. and M. Teboulle. "Smoothing and first order methods: A unified framework". In: *SIAM J. Optim.* 22.2 (2012), pp. 557–580.
- Bertsekas, D. P. *Nonlinear programming*. Athena Scientific, 1999.
- Bertsimas, D., D. B. Brown, and C. Caramanis. "Theory and applications of robust optimization". In: *SIAM Rev.* 53.3 (2011), pp. 464–501.

References II

- Boyd, S. and L. Vandenberghe. *Convex optimization*. Cambridge University Press, 2004, pp. xiv+716.
- Chen, M., C. Gao, and Z. Ren. "A General Decision Theory for Huber's ϵ-Contamination Model". In: *Electron. J. Stat.* 10.2 (2016), pp. 3752–3774.
- Chen, Y., C. Caramanis, and S. Mannor. "Robust sparse regression under adversarial corruption". In: *ICML*. 2013, pp. 774–782.
- Donoho, D. L. and M. Gasko. "Breakdown properties of location estimates based on halfspace depth and projected outlyingness". In: *Ann. Statist.* 20.4 (1992), pp. 1803–1827.
- Golub, G. H. and C. F. van Loan. *Matrix computations*. Fourth. Johns Hopkins University Press, Baltimore, MD, 2013, pp. xiv+756.
- Hampel, F. R. et al. *Robust statistics: The Approach Based on Influence Functions*. Wiley series in probability and statistics. Wiley, 1986.

References III

- Hiriart-Urruty, J.-B. and C. Lemaréchal. *Convex analysis and minimization algorithms. I*. Vol. 305. Berlin: Springer-Verlag, 1993.
- – .*Convex analysis and minimization algorithms. II*. Vol. 306. Berlin: Springer-Verlag, 1993.
- Horn, R. A. and C. R. Johnson. *Topics in matrix analysis*. Corrected reprint of the 1991 original. Cambridge: Cambridge University Press, 1994, pp. viii+607.
- Huber, P. J. and E. M. Ronchetti. *Robust statistics*. Second. Wiley series in probability and statistics. Wiley, 2009.
- Maronna, R. A., R. D. Martin, and V. J. Yohai. *Robust statistics: Theory and methods*. Chichester: John Wiley & Sons, 2006.
- Minsker, S. "Geometric median and robust estimation in Banach spaces". In: *Bernoulli* 21.4 (2015), pp. 2308–2335.
- Mosler, K. "Depth statistics". In: *Robustness and complex data structures*. Springer, 2013, pp. 17–34.

References IV

- Murphy, K. P. *Machine learning: a probabilistic perspective*. MIT press, 2012.
- Nesterov, Y. "Smooth minimization of non-smooth functions". In: *Math. Program.* 103.1 (2005), pp. 127–152.
- Parikh, N. et al. "Proximal algorithms". In: *Foundations and Trends in Machine Learning* 1.3 (2013), pp. 1–108.
- Rousseeuw, P. J. and A. M. Leroy. *Robust regression and outlier detection*. Wiley Series in Probability and Mathematical Statistics: Applied Probability and Statistics. New York: John Wiley & Sons Inc., 1987, pp. xvi+329.
- Seber, G. A. F. and A. J. Lee. *Linear Regression Analysis, 2nd edition (Wiley Series in Probability and Statistics)*. 2nd ed. Wiley, 2003.
- Wei, X. and S. Minsker. "Estimation of the covariance structure of heavy-tailed distributions". In: *NIPS*. 2017, pp. 2859–2868.

References V

- Xu, H., C. Caramanis, and S. Mannor. "Robust regression and Lasso". In: *IEEE Trans. Inf. Theory* 56.7 (2010), pp. 3561–3574.