Statistical Theory and Modeling (ST2601)

Lecture 11 - Nonlinear regression and Regularization

Mattias Villani

Department of Statistics
Stockholm University

mattiasvillani.com @matvil @matvil mattiasvillani

Overview

- **Non-linear regression**
- **Regularization**
- **Exponential growth regression**

Polynomial regression

- **Polynomial regression** of degree/order p

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \ldots + \beta_p x^p + \varepsilon, \quad \varepsilon \overset{\text{iid}}{\sim} N(0, \sigma_\varepsilon^2)$$

- **Nonlinear in x**
- **Linear in $\beta_0, \beta_1, \ldots, \beta_p$**
- Polynomial regression is just a linear regression with features:
  - $x_1 = x$
  - $x_2 = x^2$
  - $\vdots$
  - $x_p = x^p$
- Can use **least squares estimate** for the model

$$y_i = \boldsymbol{x}_i^\top \boldsymbol{\beta} + \varepsilon_i, \quad \varepsilon_i \overset{\text{iid}}{\sim} N(0, \sigma_\varepsilon^2)$$

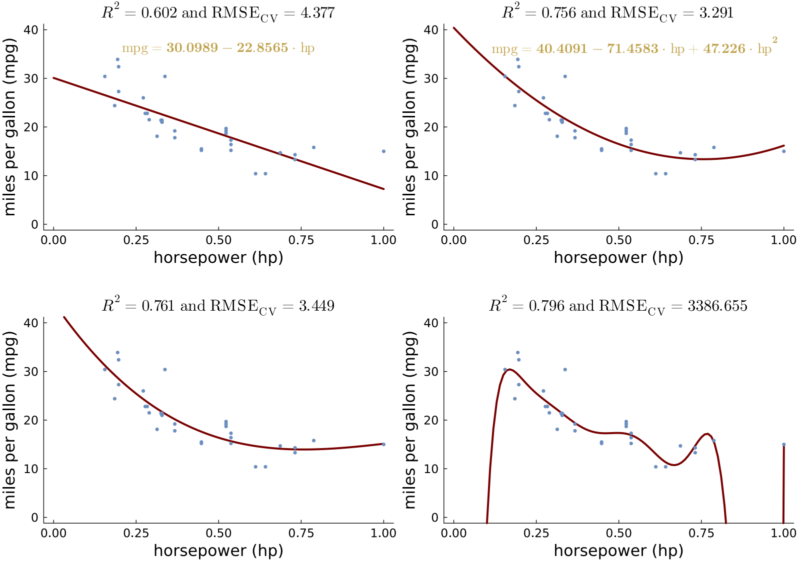
where the covariate/feature vector has $p + 1$ elements

$$\boldsymbol{x}_i = (1, x_i, x_i^2, \ldots, x_i^p)^\top$$

Polynomial regression data setup

	A	B	C	D	E	F
1		mpg (y)	hp (x)	x^2	x^3	x^4
2	Mazda RX4	21.000	0.328	0.108	0.035	0.012
3	Mazda RX4 Wag	21.000	0.328	0.108	0.035	0.012
4	Datsun 710	22.800	0.278	0.077	0.021	0.006
5	Hornet 4 Drive	21.400	0.328	0.108	0.035	0.012
6	Hornet Sportabo	18.700	0.522	0.273	0.143	0.074
7	Valiant	18.100	0.313	0.098	0.031	0.010
8	Duster 360	14.300	0.731	0.535	0.391	0.286
9	Merc 240D	24.400	0.185	0.034	0.006	0.001
10	Merc 230	22.800	0.284	0.080	0.023	0.006
11	Merc 280	19.200	0.367	0.135	0.049	0.018
12	Merc 280C	17.800	0.367	0.135	0.049	0.018
13	Merc 450SE	16.400	0.537	0.289	0.155	0.083
14	Merc 450SL	17.300	0.537	0.289	0.155	0.083
15	Merc 450SLC	15.200	0.537	0.289	0.155	0.083
16	Cadillac Fleetwo	10.400	0.612	0.374	0.229	0.140
17	Lincoln Continer	10.400	0.642	0.412	0.264	0.170
18	Chrysler Imperia	14.700	0.687	0.471	0.324	0.222
19	Fiat 128	32.400	0.197	0.039	0.008	0.002
20	Honda Civic	30.400	0.155	0.024	0.004	0.001
21	Toyota Corolla	33.900	0.194	0.038	0.007	0.001
22	Toyota Corona	21.500	0.290	0.084	0.024	0.007
23	Dodge Challeng	15.500	0.448	0.200	0.090	0.040
24	AMC Javelin	15.200	0.448	0.200	0.090	0.040
25	Camaro Z28	13.300	0.731	0.535	0.391	0.286
26	Pontiac Firebird	19.200	0.522	0.273	0.143	0.074
27	Fiat X1-9	27.300	0.197	0.039	0.008	0.002
28	Porsche 914-2	26.000	0.272	0.074	0.020	0.005
29	Lotus Europa	30.400	0.337	0.114	0.038	0.013
30	Ford Pantera L	15.800	0.788	0.621	0.489	0.386
31	Ferrari Dino	19.700	0.522	0.273	0.143	0.074

Polynomial regression for mtcars data



K-fold cross-validation

Fold 1

biltyp	mpg	hp
Hornet Sportabout	18.7	0.52
Fiat X1-9	27.3	0.20
Merc 450SL	17.3	0.54
Merc 450SLC	15.2	0.54
Merc 240D	24.4	0.19
Duster 360	14.3	0.73
Datsun 710	22.8	0.28
Ferrari Dino	19.7	0.52
Ford Pantera L	15.8	0.79
Pontiac Firebird	19.2	0.52
Toyota Corona	21.5	0.29
AMC Javelin	15.2	0.45
Camaro Z28	13.3	0.73
Fiat 128	32.4	0.20
Merc 280C	17.8	0.37
Lotus Europa	30.4	0.34
Cadillac Fleetwood	10.4	0.61
Chrysler Imperial	14.7	0.69
Mazda RX4	21	0.33
Volvo 142E	21.4	0.33
Mazda RX4 Wag	21	0.33
Merc 230	22.8	0.28
Toyota Corolla	33.9	0.19
Merc 280	19.2	0.37
Dodge Challenger	15.5	0.45
Lincoln Continental	10.4	0.64
Valiant	18.1	0.31
Honda Civic	30.4	0.16
Hornet 4 Drive	21.4	0.33
Merc 450SE	16.4	0.54
Maserati Bora	15	1.00
Porsche 914-2	26	0.27

Fold 2

biltyp	mpg	hp
Hornet Sportabout	18.7	0.52
Fiat X1-9	27.3	0.20
Merc 450SL	17.3	0.54
Merc 450SLC	15.2	0.54
Merc 240D	24.4	0.19
Duster 360	14.3	0.73
Datsun 710	22.8	0.28
Ferrari Dino	19.7	0.52
Ford Pantera L	15.8	0.79
Pontiac Firebird	19.2	0.52
Toyota Corona	21.5	0.29
AMC Javelin	15.2	0.45
Camaro Z28	13.3	0.73
Fiat 128	32.4	0.20
Merc 280C	17.8	0.37
Lotus Europa	30.4	0.34
Cadillac Fleetwood	10.4	0.61
Chrysler Imperial	14.7	0.69
Mazda RX4	21	0.33
Volvo 142E	21.4	0.33
Mazda RX4 Wag	21	0.33
Merc 230	22.8	0.28
Toyota Corolla	33.9	0.19
Merc 280	19.2	0.37
Dodge Challenger	15.5	0.45
Lincoln Continental	10.4	0.64
Valiant	18.1	0.31
Honda Civic	30.4	0.16
Hornet 4 Drive	21.4	0.33
Merc 450SE	16.4	0.54
Maserati Bora	15	1.00
Porsche 914-2	26	0.27

Fold 3

biltyp	mpg	hp
Hornet Sportabout	18.7	0.52
Fiat X1-9	27.3	0.20
Merc 450SL	17.3	0.54
Merc 450SLC	15.2	0.54
Merc 240D	24.4	0.19
Duster 360	14.3	0.73
Datsun 710	22.8	0.28
Ferrari Dino	19.7	0.52
Ford Pantera L	15.8	0.79
Pontiac Firebird	19.2	0.52
Toyota Corona	21.5	0.29
AMC Javelin	15.2	0.45
Camaro Z28	13.3	0.73
Fiat 128	32.4	0.20
Merc 280C	17.8	0.37
Lotus Europa	30.4	0.34
Cadillac Fleetwood	10.4	0.61
Chrysler Imperial	14.7	0.69
Mazda RX4	21	0.33
Volvo 142E	21.4	0.33
Mazda RX4 Wag	21	0.33
Merc 230	22.8	0.28
Toyota Corolla	33.9	0.19
Merc 280	19.2	0.37
Dodge Challenger	15.5	0.45
Lincoln Continental	10.4	0.64
Valiant	18.1	0.31
Honda Civic	30.4	0.16
Hornet 4 Drive	21.4	0.33
Merc 450SE	16.4	0.54
Maserati Bora	15	1.00
Porsche 914-2	26	0.27

Fold 4

biltyp	mpg	hp
Hornet Sportabout	18.7	0.52
Fiat X1-9	27.3	0.20
Merc 450SL	17.3	0.54
Merc 450SLC	15.2	0.54
Merc 240D	24.4	0.19
Duster 360	14.3	0.73
Datsun 710	22.8	0.28
Ferrari Dino	19.7	0.52
Ford Pantera L	15.8	0.79
Pontiac Firebird	19.2	0.52
Toyota Corona	21.5	0.29
AMC Javelin	15.2	0.45
Camaro Z28	13.3	0.73
Fiat 128	32.4	0.20
Merc 280C	17.8	0.37
Lotus Europa	30.4	0.34
Cadillac Fleetwood	10.4	0.61
Chrysler Imperial	14.7	0.69
Mazda RX4	21	0.33
Volvo 142E	21.4	0.33
Mazda RX4 Wag	21	0.33
Merc 230	22.8	0.28
Toyota Corolla	33.9	0.19
Merc 280	19.2	0.37
Dodge Challenger	15.5	0.45
Lincoln Continental	10.4	0.64
Valiant	18.1	0.31
Honda Civic	30.4	0.16
Hornet 4 Drive	21.4	0.33
Merc 450SE	16.4	0.54
Maserati Bora	15	1.00
Porsche 914-2	26	0.27

- Fold k:
  - Index for **test observations** in fold k: $\mathcal{T}_k$.
  - Model is fitted to **training data** in fold k
  - Predictions $\hat{y}_i^{(k)}$ for test data $i \in \mathcal{T}_k$.

K-fold cross-validation

Fold 1

biltyp	mpg	hp
Hornet Sportabout	18.7	0.52
Fiat X1-9	27.3	0.20
Merc 450SL	17.3	0.54
Merc 450SLC	15.2	0.54
Merc 240D	24.4	0.19
Duster 360	14.3	0.73
Datsun 710	22.8	0.28
Ferrari Dino	19.7	0.52
Ford Pantera L	15.8	0.79
Pontiac Firebird	19.2	0.52
Toyota Corona	21.5	0.29
AMC Javelin	15.2	0.45
Camaro Z28	13.3	0.73
Fiat 128	32.4	0.20
Merc 280C	17.8	0.37
Lotus Europa	30.4	0.34
Cadillac Fleetwood	10.4	0.61
Chrysler Imperial	14.7	0.69
Mazda RX4	21	0.33
Volvo 142E	21.4	0.33
Mazda RX4 Wag	21	0.33
Merc 230	22.8	0.28
Toyota Corolla	33.9	0.19
Merc 280	19.2	0.37
Dodge Challenger	15.5	0.45
Lincoln Continental	10.4	0.64
Valiant	18.1	0.31
Honda Civic	30.4	0.16
Hornet 4 Drive	21.4	0.33
Merc 450SE	16.4	0.54
Maserati Bora	15	1.00
Porsche 914-2	26	0.27

Fold 2

biltyp	mpg	hp
Hornet Sportabout	18.7	0.52
Fiat X1-9	27.3	0.20
Merc 450SL	17.3	0.54
Merc 450SLC	15.2	0.54
Merc 240D	24.4	0.19
Duster 360	14.3	0.73
Datsun 710	22.8	0.28
Ferrari Dino	19.7	0.52
Ford Pantera L	15.8	0.79
Pontiac Firebird	19.2	0.52
Toyota Corona	21.5	0.29
AMC Javelin	15.2	0.45
Camaro Z28	13.3	0.73
Fiat 128	32.4	0.20
Merc 280C	17.8	0.37
Lotus Europa	30.4	0.34
Cadillac Fleetwood	10.4	0.61
Chrysler Imperial	14.7	0.69
Mazda RX4	21	0.33
Volvo 142E	21.4	0.33
Mazda RX4 Wag	21	0.33
Merc 230	22.8	0.28
Toyota Corolla	33.9	0.19
Merc 280	19.2	0.37
Dodge Challenger	15.5	0.45
Lincoln Continental	10.4	0.64
Valiant	18.1	0.31
Honda Civic	30.4	0.16
Hornet 4 Drive	21.4	0.33
Merc 450SE	16.4	0.54
Maserati Bora	15	1.00
Porsche 914-2	26	0.27

Fold 3

biltyp	mpg	hp
Hornet Sportabout	18.7	0.52
Fiat X1-9	27.3	0.20
Merc 450SL	17.3	0.54
Merc 450SLC	15.2	0.54
Merc 240D	24.4	0.19
Duster 360	14.3	0.73
Datsun 710	22.8	0.28
Ferrari Dino	19.7	0.52
Ford Pantera L	15.8	0.79
Pontiac Firebird	19.2	0.52
Toyota Corona	21.5	0.29
AMC Javelin	15.2	0.45
Camaro Z28	13.3	0.73
Fiat 128	32.4	0.20
Merc 280C	17.8	0.37
Lotus Europa	30.4	0.34
Cadillac Fleetwood	10.4	0.61
Chrysler Imperial	14.7	0.69
Mazda RX4	21	0.33
Volvo 142E	21.4	0.33
Mazda RX4 Wag	21	0.33
Merc 230	22.8	0.28
Toyota Corolla	33.9	0.19
Merc 280	19.2	0.37
Dodge Challenger	15.5	0.45
Lincoln Continental	10.4	0.64
Valiant	18.1	0.31
Honda Civic	30.4	0.16
Hornet 4 Drive	21.4	0.33
Merc 450SE	16.4	0.54
Maserati Bora	15	1.00
Porsche 914-2	26	0.27

Fold 4

biltyp	mpg	hp
Hornet Sportabout	18.7	0.52
Fiat X1-9	27.3	0.20
Merc 450SL	17.3	0.54
Merc 450SLC	15.2	0.54
Merc 240D	24.4	0.19
Duster 360	14.3	0.73
Datsun 710	22.8	0.28
Ferrari Dino	19.7	0.52
Ford Pantera L	15.8	0.79
Pontiac Firebird	19.2	0.52
Toyota Corona	21.5	0.29
AMC Javelin	15.2	0.45
Camaro Z28	13.3	0.73
Fiat 128	32.4	0.20
Merc 280C	17.8	0.37
Lotus Europa	30.4	0.34
Cadillac Fleetwood	10.4	0.61
Chrysler Imperial	14.7	0.69
Mazda RX4	21	0.33
Volvo 142E	21.4	0.33
Mazda RX4 Wag	21	0.33
Merc 230	22.8	0.28
Toyota Corolla	33.9	0.19
Merc 280	19.2	0.37
Dodge Challenger	15.5	0.45
Lincoln Continental	10.4	0.64
Valiant	18.1	0.31
Honda Civic	30.4	0.16
Hornet 4 Drive	21.4	0.33
Merc 450SE	16.4	0.54
Maserati Bora	15	1.00
Porsche 914-2	26	0.27

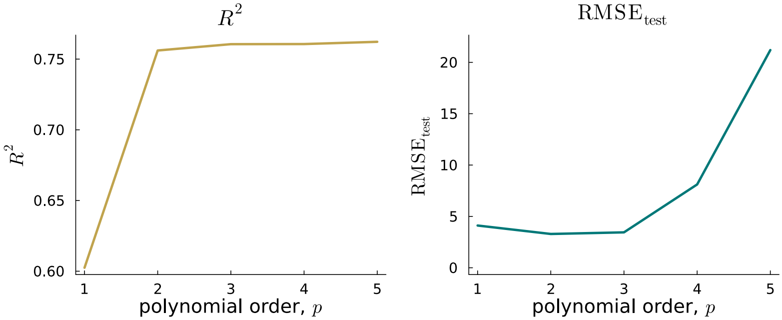
- **K-fold cross-validated prediction error**

$$\mathrm{SSE}_{CV} = \sum_{i \in \mathcal{T}_1} \left(y_i - \hat{y}_i^{(1)} \right)^2 + \ldots + \sum_{i \in \mathcal{T}_K} \left(y_i - \hat{y}_i^{(K)} \right)^2$$

$$\mathrm{RMSE}_{CV} = \sqrt{\frac{\mathrm{SSE}_{CV}}{n}}$$

- Can be used for **model choice**, for example polynomial order.

mtcars data - R^2 and RMSE-CV ($K = 4$)



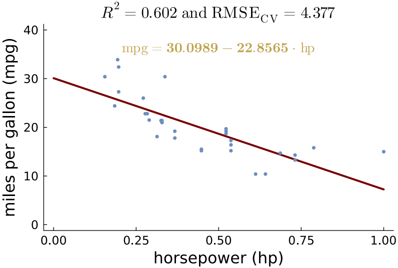
Interpretation in nonlinear model is more tricky

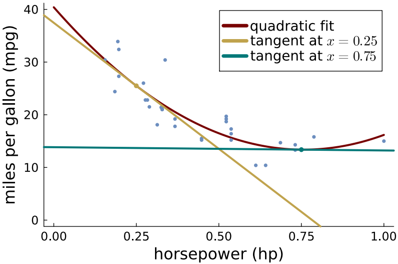
- **Derivative**: how much does y change when x changes?
- **Linear** model - derivative does not depend on x

$$\frac{\mathrm{d}}{\mathrm{d}x}(\beta_0 + \beta_1 x) = \beta_1$$

- **Quadratic** model - derivative depends on x

$$\frac{\mathrm{d}}{\mathrm{d}x}(\beta_0 + \beta_1 x + \beta_2 x^2) = \beta_1 + 2\beta_2 x$$





L2-regularization (Ridge regression)

- Least squares **minimizes residual sum of squares**

$$\mathrm{RSS}(\beta_0, \beta_1) = \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i)^2$$

- Same estimator as from **maximum likelihood**

$$\ell(\beta_0, \beta_1) = -\frac{n}{2} \log(2\pi\sigma^2) - \frac{1}{2\sigma_\varepsilon^2} \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i)^2$$

- Flexible models with many parameters can **overfit**.
- **Regularization** penalizes large values of the parameters.
- **L2-regularization**

$$\mathrm{RSS}_{\mathrm{P}}(\beta_0, \beta_1) = \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i)^2 \quad + \quad \underbrace{\lambda \cdot (\beta_0^2 + \beta_1^2)}_{\text{L2-penalty}}$$

L2-regularization (Ridge regression)

- Multiple regression: least squares $\hat{\boldsymbol{\beta}} = (\boldsymbol{X}^\top \boldsymbol{X})^{-1} \boldsymbol{X}^\top \boldsymbol{y}$ minimizes

$$\mathrm{RSS}(\boldsymbol{\beta}) = \sum_{i=1}^{n} (y_i - \boldsymbol{x}_i^\top \boldsymbol{\beta})^2 = (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})^\top (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})$$

- **L2-regularization**

$$\mathrm{RSS_P}(\boldsymbol{\beta}) = (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})^\top (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}) \quad + \quad \underbrace{\lambda \cdot \boldsymbol{\beta}^\top \boldsymbol{\beta}}_{\text{L2−penalty}}$$

- Solving for $\boldsymbol{\beta}$ (Linear Algebra section in the Prequel book)

$$\frac{\partial}{\partial \boldsymbol{\beta}} \mathrm{RSS_P}(\boldsymbol{\beta}) = -2\boldsymbol{X}^\top (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}) + 2\lambda \boldsymbol{\beta} = \mathbf{0}$$

$$\text{solution:} \quad \hat{\boldsymbol{\beta}}_{L_2} = (\boldsymbol{X}^\top \boldsymbol{X} + \lambda \boldsymbol{I}_p)^{-1} \boldsymbol{X}^\top \boldsymbol{y}$$

- **Shrinkage** of least squares $\hat{\boldsymbol{\beta}}$ toward zero. When $\boldsymbol{X}^\top \boldsymbol{X} = I_p$,

$$\hat{\boldsymbol{\beta}}_{L_2} = \frac{1}{1+\lambda} \hat{\boldsymbol{\beta}}$$

L1-regularization (Lasso regression)

- **L1-regularization** (**Lasso**)

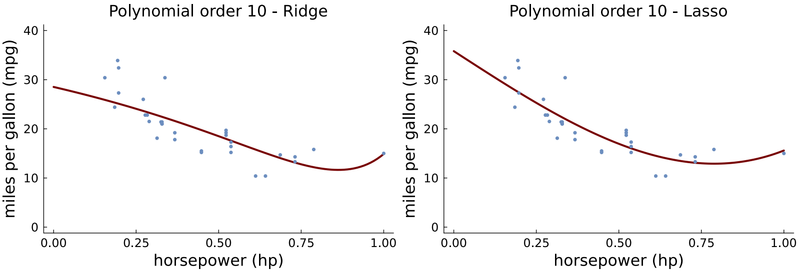
$$\mathrm{RSS}_{\mathrm{P}}(\boldsymbol{\beta}) = (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta})^\top (\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}) \quad + \quad \underbrace{\lambda \cdot \sum_{j=1}^{p} \mid \beta_j \mid}_{\text{L1}-\text{penalty}}$$

- No explicit formula, but very efficient algorithm (LARS).
- Lasso does both:
  - **shrinkage** and
  - **selection** - sets some $\hat{\beta}_j$ exactly to zero.
- $L1$ and $L2$ regularization can be seen a **Bayesian priors**. 😍

Regularization mtcars data

- Shrinkage parameter λ selected by cross-validation.
- Lasso:

$$y = 35.81 - 43.54 \cdot \mathrm{hp} + 23.32 \cdot \mathrm{hp}^3$$



Exponential (growth) regression

- **Model**:

$$Y = \beta_0 \cdot \beta_1^x \cdot \varepsilon, \quad \varepsilon \sim \mathrm{LogNormal}(0, \sigma_\varepsilon^2)$$

- Take **logs** to make the model **linear**!

$$\underbrace{\log Y}_{\tilde{y}} = \underbrace{\log \beta_0}_{\gamma_0} + \underbrace{\log \beta_1}_{\gamma_1} \cdot x + \underbrace{\log \varepsilon}_{\tilde{\varepsilon}}$$

$$\tilde{Y} = \gamma_0 + \gamma_1 \cdot x + \tilde{\varepsilon}, \qquad \tilde{\varepsilon} \sim N\left(0, \sigma_{\tilde{\varepsilon}}^2\right).$$

- Exponential regression can be **fit by least squares on log y**!
- **Prediction** at $x = x^\star$:
  - Predict $\tilde{y}$ on the log scale
  - Transform to original scale: $e^{\tilde{y}}$

Chinese growth

	A	B	C	D	E
1	year	gdp	gdpgrowth	log10(gdp)	t = year - 1999
2	2000	959.3725	9.86	2.981987265	1
3	2001	1053.1082	9.77	3.022472994	2
4	2002	1148.5083	9.06	3.060134138	3
5	2003	1288.6433	12.2	3.11013272	4
6	2004	1508.6681	17.07	3.178593708	5
7	2005	1753.4178	16.22	3.243885411	6
8	2006	2099.2294	19.72	3.3220599	7
9	2007	2693.9701	28.33	3.430392771	8
10	2008	3468.3046	28.74	3.540117232	9
11	2009	3832.2364	10.49	3.583452292	10
12	2010	4550.4531	18.74	3.658054643	11
13	2011	5618.1323	23.46	3.749591962	12
14	2012	6316.9183	12.44	3.80050526	13
15	2013	7050.6463	11.62	3.848228929	14
16	2014	7678.5995	8.91	3.885282016	15
17	2015	8066.9426	5.06	3.906708967	16
18	2016	8147.9377	1	3.9110477	17
19	2017	8879.4387	8.98	3.948385513	18
20	2018	9976.6771	12.36	3.998985916	19
21	2019	10216.6303	2.41	4.009307678	20
22	2020	10500.3956	2.78	4.021205661	21
23					

Chinese growth 2000-2013

- y = growth GDP (gross domestic product)
- x = year-1999 (so $x = 1$ is the year 2000)

Coefficients:

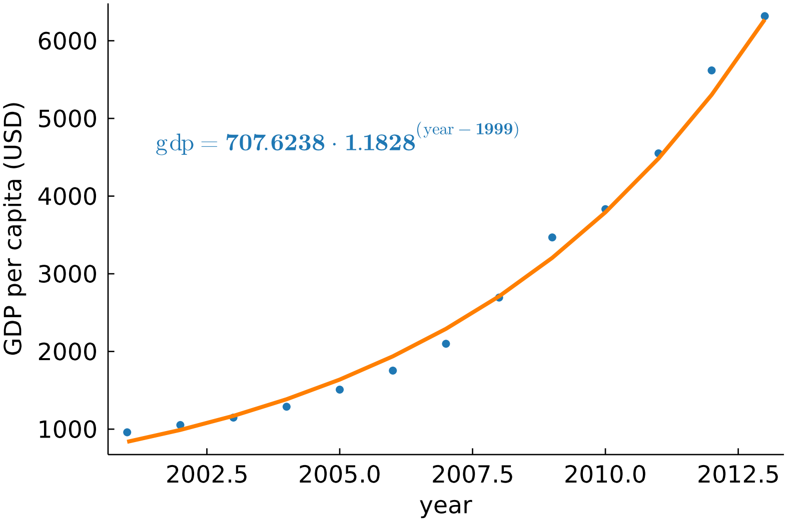
	Coef.	Std. Error	t	Pr(>|t|)	Lower 95%	Upper 95%
(Intercept)	2.8498	0.0192341	148.16	<1e-18	2.80747	2.89214
year	0.0729005	0.00242327	30.08	<1e-11	0.067567	0.0782341

- $\hat{\gamma}_0 = 2.8498$, so $\hat{\beta}_0 = 10^{\hat{\gamma}_0} = 10^{2.8498} \approx 707.62$.
- $\hat{\gamma}_1 = 0.0729$, so $\hat{\beta}_1 = 10^{\hat{\gamma}_1} = 10^{0.0729005} = 1.18277$.
- Fitted model on original scale

$$\hat{y} = \hat{\beta}_0 \cdot \hat{\beta}_1^x = 707.62 \cdot 1.18277^x$$

- 18% yearly growth!

Chinese growth 2000-2013



Chinese growth 2000-2021