Regression

1. Solving Overdetermined Linear Systems
   - numerical linear algebra
   - fitting numerical data
   - fitting exponential growth by linearization
2. Predictive Models
   - ordinary least squares
   - fitting average temperatures
   - regression as a data mining task
3. Proposals of Project Topics
   - fit processor data – fit census data
   - fit corona virus data
   - fit student performance data

MCS 472 Lecture 16
Industrial Math & Computation
Jan Verschelde, 18 February 2026

Regression

1. Solving Overdetermined Linear Systems
   - numerical linear algebra
   - fitting numerical data
   - fitting exponential growth by linearization
2. Predictive Models
   - ordinary least squares
   - fitting average temperatures
   - regression as a data mining task
3. Proposals of Project Topics
   - fit processor data – fit census data
   - fit corona virus data
   - fit student performance data

solving overdetermined linear systems

Consider $A\mathbf{x} = \mathbf{b}$, A is m-by-n, with $m \geq n$.

How to solve this overdetermined linear system?

- Minimize $\|\mathbf{b} - A\mathbf{x}\|_2^2$.
- The Householder QR is numerically stable.
- The Generalized Minimum Residual Method is iterative.
- Orthogonality: $\langle \mathbf{b} - A\mathbf{x}, A \rangle = 0$.
- Fit with polynomials or exponentials.

a Julia session

The backslash operator applies to overdetermined systems.

```
julia> A = rand(3,2); b = rand(3,1);

julia> x = A\b;

julia> r = b - A*x
3×1 Matrix{Float64}:
 -0.038628108251110294
  0.4381284983642053
 -0.17773235808476184

julia> r'*A
1×2 Matrix{Float64}:
 -2.77556e-17  -2.77556e-17
```

The outcome of `r'*A` or $\langle \mathbf{b} - A\mathbf{x}, A \rangle$ is below machine precision.

Regression

1. Solving Overdetermined Linear Systems
   - numerical linear algebra
   - fitting numerical data
   - fitting exponential growth by linearization
2. Predictive Models
   - ordinary least squares
   - fitting average temperatures
   - regression as a data mining task
3. Proposals of Project Topics
   - fit processor data – fit census data
   - fit corona virus data
   - fit student performance data

fitting numerical data

Find a trend in numerical data.

Input: (x_i, y_i), $i = 1, 2, \ldots, m$.

Output: (a, b), slope and intercept of a line $y = ax + b$,

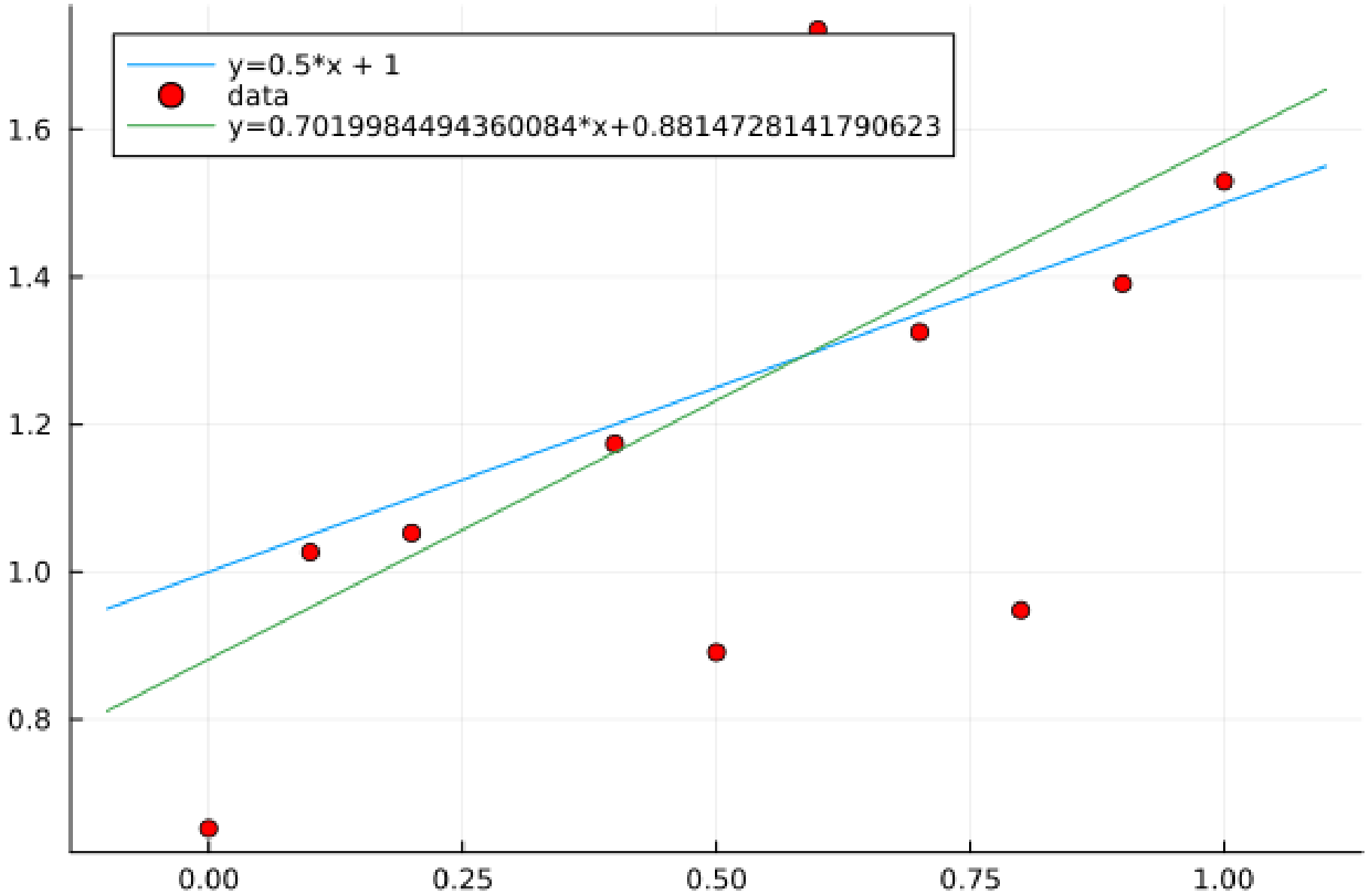
so that

$$\sum_{i=1}^{m} \Big(y_i - (ax_i + b) \Big)^2$$

is minimal.

If the slope a is positive, then the trend in the data is increasing.

fitting noisy data



the Julia code in the Jupyter notebook

```
using Plots
# plot the exact data on the line y = 0.5*x + 1
x = -0.1:0.1:1.1
y = 0.5*x .+ 1
plot(x,y,label="y=0.5*x + 1",legend=:topleft)
# add normally distributed noise
ynoisy = [y[i] + 0.2*randn() for i=1:length(y)]
scatter!(x[2:length(x)-1],ynoisy[2:length(x)-1],
         marker=(:red),label="data",legend=:topleft)
# set up the linear system
A = ones(length(x),2)
A[:,2] = x
# apply the backslash operator
c = A\ynoisy
# plot the fitted line
lbl = string("y=", string(c[2]), "*x+", string(c[1]))
fity = c[1] .+ c[2]*x
plot!(x,fity,label=lbl)
```

Regression

1. Solving Overdetermined Linear Systems
   - numerical linear algebra
   - fitting numerical data
   - fitting exponential growth by linearization
2. Predictive Models
   - ordinary least squares
   - fitting average temperatures
   - regression as a data mining task
3. Proposals of Project Topics
   - fit processor data – fit census data
   - fit corona virus data
   - fit student performance data

fitting exponential growth by linearization

Given (t_i, f_i), $i = 0, 1, \ldots, n$, find the coefficients of

$$y(t) = c_1 e^{c_2 t} = c_1 \exp(c_2 t),$$

so $y(t)$ fits the data (t_i, f_i) best.

With linearization, we consider

$$\ln(y(t)) = \ln(c_1 e^{c_2 t}) = \ln(c_1) + c_2 t.$$

The logarithmic scale helps to deal with numerical instabilities caused by extreme values which may arise with exponential growth.

on Moore's Law

year	Intel processor	transistor count
2000	Pentium III Coppermine	21,000,000
2001	Pentium III Tualatin	45,000,000
2002	Pentium IV Northwood	55,000,000
2003	Itanium 2 Madison	410,000,000
2004	Itanium 2	592,000,000
2005	Pentium D Smithfield	228,000,000
2006	Dual-core Itanium 2	1,700,000,000
2007	Core 2 Duo Wolfdale	411,000,000
2008	Core i7 quad-core	731,000,000
2010	Xeon Nehalem-EX (8-core)	2,300,000,000

source: `https://en.wikipedia.org/wiki/Transistor_count`

Exercise 1: Fit the above data to $y(t) = c_1 2^{c_2 t}$.
Make a $\log_2$ scale plot: (year, $\log_2$(transistor count)) of the data, and the model $y(t)$. (*For year, use 0, 1, ..., 10.*)

salary versus experience

How does salary relate to experience? Look at a case study.

Exercise 2:
Visit the ISBE Data Library of `www.illinoisreportcard.com`, download the 2025 Report Card Public Data Set (`xlsx` file).

Consider two vectors of data:

- x the average teaching experience, and
- y the average teacher salary.

Do a linear fit on (x, y) to model salary on experience.

Regression

1. Solving Overdetermined Linear Systems
   - numerical linear algebra
   - fitting numerical data
   - fitting exponential growth by linearization
2. Predictive Models
   - ordinary least squares
   - fitting average temperatures
   - regression as a data mining task
3. Proposals of Project Topics
   - fit processor data – fit census data
   - fit corona virus data
   - fit student performance data

ordinary least squares

The Generalized Linear Model is abbreviated as GLM.

An example from the GLM documentation.

```
julia> using DataFrames, GLM

julia> data = DataFrame(X=[1,2,3], Y=[2,4,7])
3x2 DataFrame
 Row  X      Y
      Int64  Int64

   1      1      2
   2      2      4
   3      3      7
```

defining a linear model

```
julia> ols = lm(@formula(Y ~ X), data)
StatsModels.TableRegressionModel{
LinearModel{GLM.LmResp{Vector{Float64}},
GLM.DensePredChol{Float64,
LinearAlgebra.CholeskyPivoted{Float64,
Matrix{Float64}}}}, Matrix{Float64}}

Y ~ 1 + X

Coefficients:

                  Coef.  Std. Error

(Intercept)  -0.666667     0.62361
X             2.5          0.288675
```

predicting the data

```
julia> predict(ols)
3-element Vector{Float64}:
 1.8333333333333308
 4.333333333333334
 6.833333333333336

julia> line(x) = -2/3 + 2.5*x
line (generic function with 1 method)

julia> [line(t) for t in [1,2,3]]
3-element Vector{Float64}:
 1.8333333333333335
 4.333333333333333
 6.833333333333333
```

Regression

1. Solving Overdetermined Linear Systems
   - numerical linear algebra
   - fitting numerical data
   - fitting exponential growth by linearization
2. Predictive Models
   - ordinary least squares
   - fitting average temperatures
   - regression as a data mining task
3. Proposals of Project Topics
   - fit processor data – fit census data
   - fit corona virus data
   - fit student performance data

fitting average temperatures

Recall lecture 5 on analyzing data.

Are the winters in Chicago becoming milder?

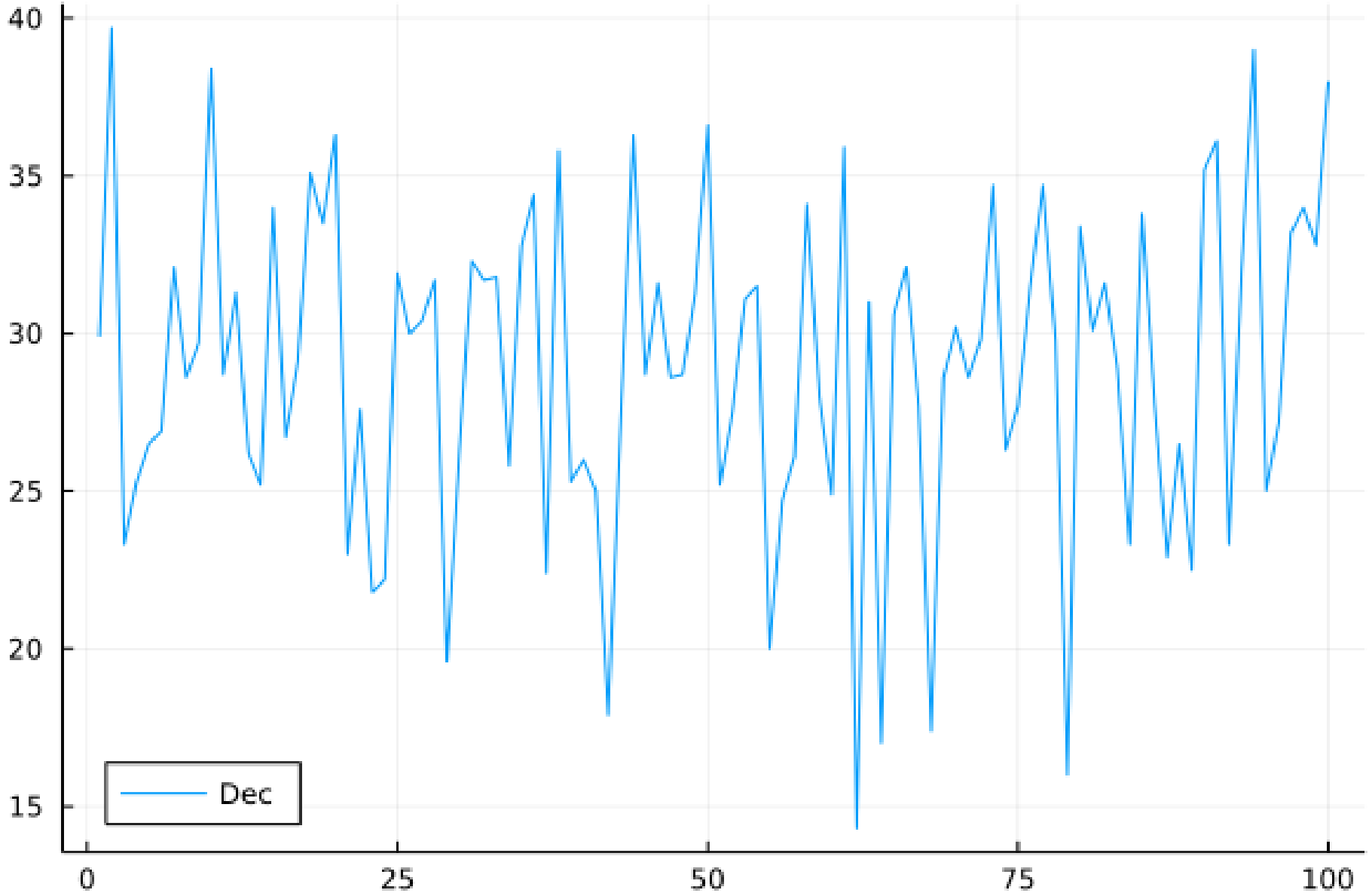
1. Data obtained from the National Weather Service, a site from the National Oceanic and Atmospheric Administration (NOAA).
2. The data are average monthly temperatures in Chicago over the past 100 years, from 1922 to 2021.
3. The file `tempChicago100years.txt` stores the data.

getting the data into a Julia session

Code in the Jupyter notebook:

```
using DelimitedFiles

A = readdlm("tempChicago100years.txt")

dectemp = A[2:end,13]

plot(1:100, dectemp, label="Dec",
     legend=:bottomleft)
```

average December temperatures in Chicago



preparing the data for the fit

Code in the Jupyter notebook continues:

```
years = A[2:end,1]

tempdf = DataFrame(X=Vector{Float64}(years),
                   Y=Vector{Float64}(dectemp))
```

The type of the elements in the matrix `A` is `Any`,
which then causes problems later, so we must convert to `Float64`.

fitting the average December temperatures

```
olstemp = lm(@formula(Y ~ X), tempdf)
```

with output:

```
Y ~ 1 + X

Coefficients:
------------------------------------
                  Coef.   Std. Error
------------------------------------
(Intercept)   20.4113       35.8233
X              0.00433003    0.0181687
```

The slope `0.00433` is tiny ...

plotting the predictions

We grab the coefficients of the fit:

```
cff = coef(olstemp)
```

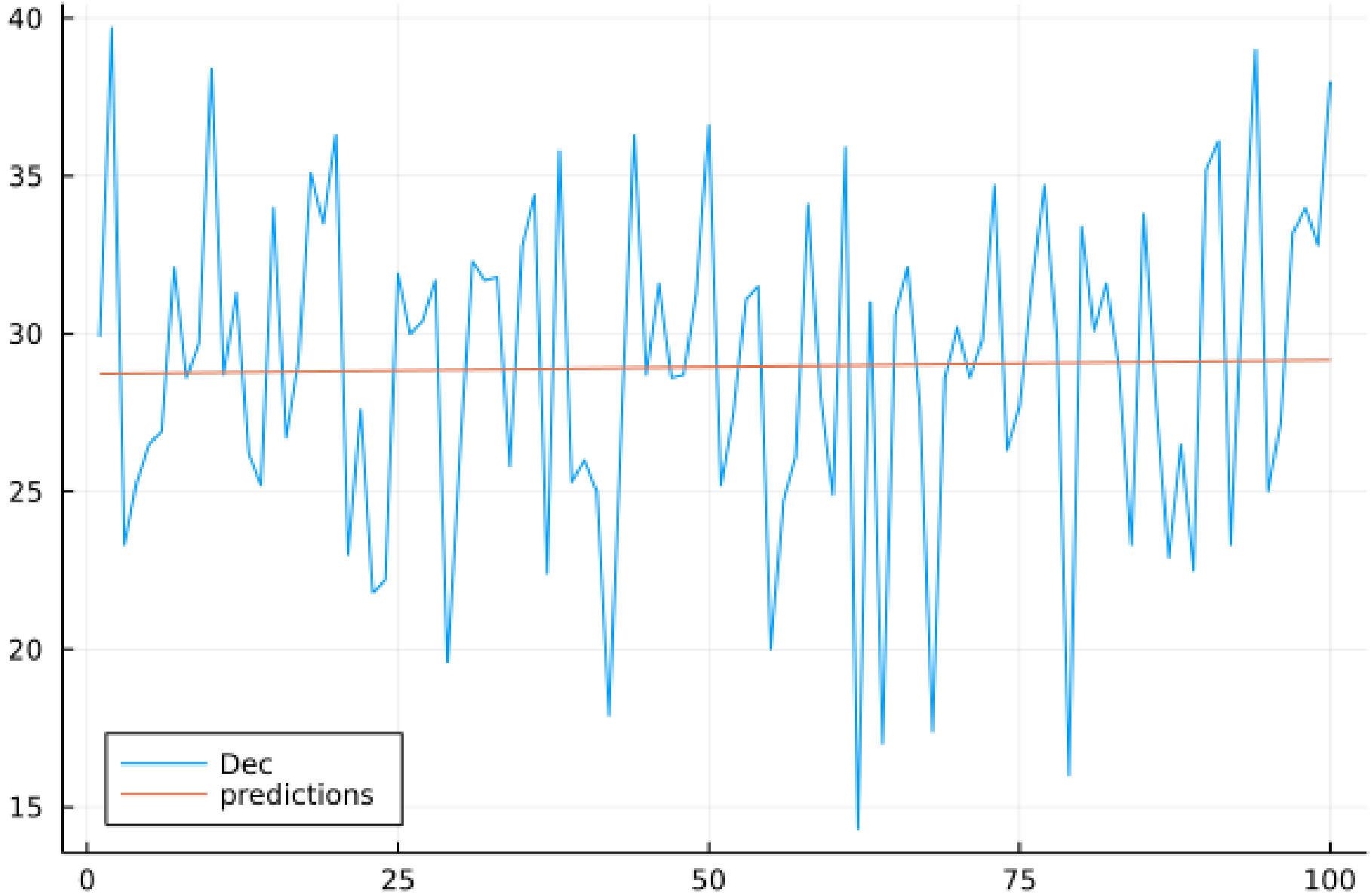
and then compute the predictions:

```
predicted = [cff[1] + cff[2]*t for t in 1922:2021]
```

and plot for the same range `1:100` as the December temperatures:

```
plot!(1:100, predicted, label="predictions")
```

fitted average December temperatures in Chicago



let us look at the last 20 years

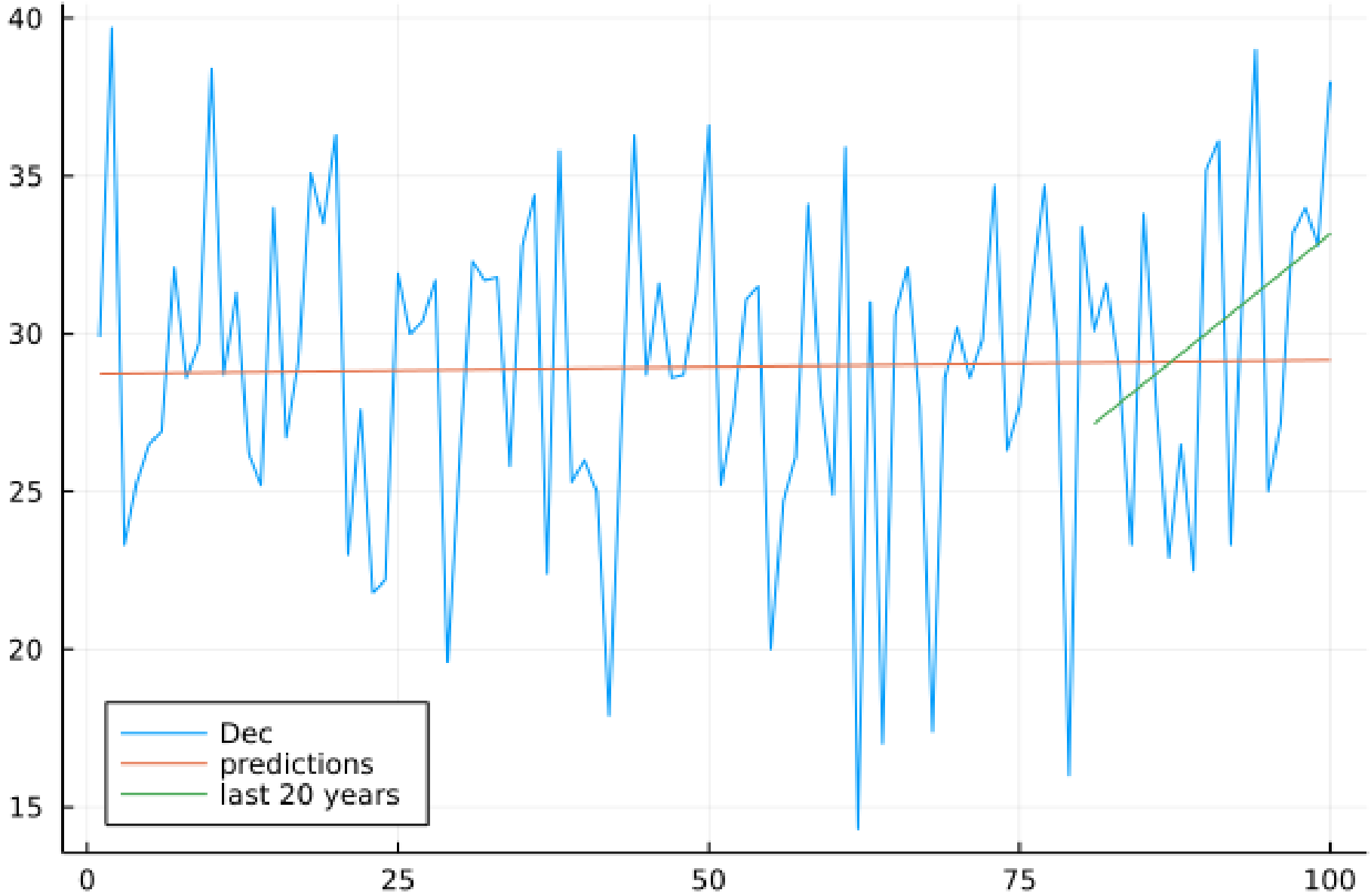
```
Xlast20=Vector{Float64}(years[end-19:end])
Ylast20=Vector{Float64}(dectemp[end-19:end])
tempdf20 = DataFrame(X=Xlast20,Y=Ylast20)

olstemp20 = lm(@formula(Y ~ X), tempdf20)

cff20 = coef(olstemp20)
```

The coefficients are

```
-607.1678947233604
   0.316842105256456
```

The larger slope `0.316` is noticeable in the plot.

fitted last 20 years



break up the data in many year segments

Exercise 3:
Complete the plot with the fit for the last 20 years
with the fits for 1922-1941, 1942-1961, 1962-1981, and 1982-2001.

Exercise 4:
Adjust the code of the previous exercise, so that instead of 20,
any divisor of 100 could be used to break up the data.

Regression

1. Solving Overdetermined Linear Systems
   - numerical linear algebra
   - fitting numerical data
   - fitting exponential growth by linearization
2. Predictive Models
   - ordinary least squares
   - fitting average temperatures
   - regression as a data mining task
3. Proposals of Project Topics
   - fit processor data – fit census data
   - fit corona virus data
   - fit student performance data

a modeling view on data mining

Definition

In the modeling view of data mining, *regression* is a task of predicting a numeric target attribute which represents some quantity of interest.

This quantity of interest could be

- an outcome or a parameter of an industrial process,
- an amount of money earned or spent,
- a cost or gain due to a business decision, etc.

Data mining has its origins in machine learning and statistics.

In data mining: *domain*, *instance*, *attribute*, and *dataset*.

In statistics: *population*, *observation*, *variable*, and *sample* are the respective counterparts to the data mining terms.

bibliography

- Jose Storopoli, Rik Huijzer, Lazaro Alonso: *Julia Data Science*. First edition published 2021. `https://juliadatascience.io`
  Creative Commons Attribution-Noncommercial-ShareAlike 4.0 International
- Pawel Cichosz: *Data Mining Algorithms: Explained Using R*. Wiley 2015.

Regression

1. Solving Overdetermined Linear Systems
   - numerical linear algebra
   - fitting numerical data
   - fitting exponential growth by linearization
2. Predictive Models
   - ordinary least squares
   - fitting average temperatures
   - regression as a data mining task
3. Proposals of Project Topics
   - fit processor data – fit census data
   - fit corona virus data
   - fit student performance data

1. fit processor data

At `https://www.cpubenchmark.net`
we can find data for over a million benchmarked processors.

Consider the following questions:

1. Does Moore's Law still hold?
2. What is the relation between price and performance?

2. fit census data

Using census data from 1790 to today,
find experimentally a best fitting polynomial for the
U.S. population in millions against time in decades.

Consider the following questions:

1. What is the best degree of polynomial?
2. Instead of one single polynomial,
   would a piecewise polynomial model fit better?

Regression

1. Solving Overdetermined Linear Systems
   - numerical linear algebra
   - fitting numerical data
   - fitting exponential growth by linearization
2. Predictive Models
   - ordinary least squares
   - fitting average temperatures
   - regression as a data mining task
3. Proposals of Project Topics
   - fit processor data – fit census data
   - fit corona virus data
   - fit student performance data

3. fit corona virus data

Use data from the covid-19 pandemic
to model the exponential growth in a surge of infections.

Consider the following questions:

1. What is the exponent in the surge?
2. Do different surges have different exponents?

Regression

1. Solving Overdetermined Linear Systems
   - numerical linear algebra
   - fitting numerical data
   - fitting exponential growth by linearization
2. Predictive Models
   - ordinary least squares
   - fitting average temperatures
   - regression as a data mining task
3. Proposals of Project Topics
   - fit processor data – fit census data
   - fit corona virus data
   - fit student performance data

4. fit student performance data

Use the Illinois Report Card Data to relate student performance to the expenditures per pupil.

Consider the following questions:

- Does per pupil expenditures predict the graduation rate?
- How does teaching experience relate to student performance?