Lecture 16: Least squares (class)

Admin:

Natural problems for solving linear systems:

1. Solve $Ax=b$ for x
2. Assuming $Ax=b$ has a solution, find the shortest solution x. (Today)
3. "Least squares": If $Ax=b$ has no solution ($b \notin R(A)$), find an x that minimizes $\|Ax-b\|$.
4. "Compressed sensing": Find the sparsest x such that $Ax=b$.

Simple application of SVD:

Theorem: For any matrix A,

- $\text{rank}(A^TA) = \text{rank}(A)$ ✓
- $R(A^TA) = R(A^T)$ ✓ $\quad R(AA^T)=R(A)$
- $N(A^TA) = N(A)$ ✓ $\quad N(AA^T)=N(A^T)$

SVD:

$$A = \sum_j \sigma_j \vec{v}_j \vec{u}_j^T \qquad A\vec{x} = \sum_j \sigma_j \vec{v}_j (u_j \cdot x)$$

for $x = \frac{\vec{u}_j}{\sigma_j}$, $A\vec{x} = \vec{v}_j$

$$R(A) = \text{Span}\{\vec{v}_j \mid \sigma_j > 0\}$$

$$R(A^T) = \text{Span}\{\vec{u}_j \mid \sigma_j > 0\}$$

$$\text{Rank}(A) = \dim R(A) = \dim R(A^T) = \#\{j \mid \sigma_j > 0\}.$$

$$A^TA = \sum_{jk} \sigma_j \sigma_k u_j v_j^T v_k u_k^T = \sum_j \sigma_j^2 u_j u_j^T \qquad R(A^TA) = \text{Span}\{u_j \mid \sigma_j^2 \neq 0\} = \text{Span}\{u_j \mid \sigma_j \neq 0\} = R(A^T)$$

in the $\{u_j\}$'s basis $[A^TA] = \begin{pmatrix} \sigma_1^2 & & \\ & \sigma_2^2 & \\ & & 0 \end{pmatrix}$

let $U = \sum_j u_j e_j^T$ (unitary) $\Rightarrow U^TA^TAU = \sum_j \sigma_j^2 e_j e_j^T = \begin{pmatrix} \sigma_1^2 & & \\ & \sigma_2^2 & \\ & & 0 \end{pmatrix}$

$A^TA\vec{u}_j = \sigma_j^2 \vec{u}_j$ (σ_j^2: e-values, $\vec{u}_j$: e-vectors)

$\Rightarrow$ to find the SVD for A, you can find the spectral decomposition of A^TA or of $AA^T = \sum_j \sigma_j^2 \vec{v}_j \vec{v}_j^T$

Example: Let

$$A = \begin{pmatrix} -2 & 1 & & & 1 \\ 1 & -2 & 1 & & \\ & 1 & -2 & 1 & \\ & & 1 & -2 & 1 \\ 1 & & & 1 & -2 \end{pmatrix} = -2I + \begin{pmatrix} 0 & 1 & & & 1 \\ 1 & 0 & 1 & & \\ & 1 & 0 & 1 & \\ & & 1 & 0 & 1 \\ 1 & & & 1 & 0 \end{pmatrix} + P^T$$

What is $\|A\|$? ≈ 4?

Why? $\quad 3.99 \leq \|A\| \leq 4$

$$\|A\| \leq \|-2I\| + \|P\| + \|P^T\| = 2\|I\| + 2\|P\| = 4$$

$$\|A\| \geq \frac{\|A\vec{x}\|}{\|\vec{x}\|} \qquad A\vec{x} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} = \vec{0}$$

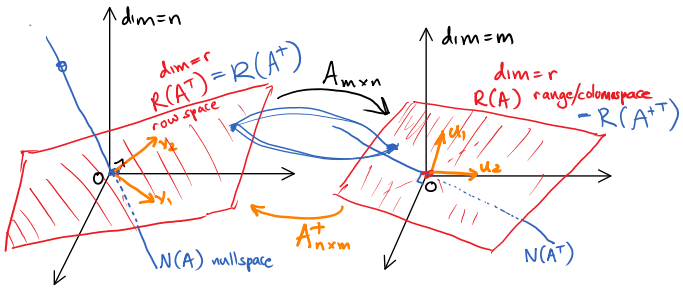
$\|A\| \le \|-2I\| + \|P\| + \|P^{-1}\| - 2\| + \|$

$\vec{x} = \begin{pmatrix} 1 \\ -1 \\ 1 \\ -1 \\ \vdots \end{pmatrix}$ Moore-Penrose $\quad \|A\| \ge \frac{\|A\vec{x}\|}{\|x\|} \quad A\vec{x} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} = \vec{0}$

Definition: Matrix pseudoinverse

$\begin{pmatrix} 2 & 0 \\ 0 & 1/3 \end{pmatrix}^{-1} = \begin{pmatrix} 1/2 & 0 \\ 0 & 3 \end{pmatrix}$

$\begin{pmatrix} 2 & 0 \\ 0 & 0 \end{pmatrix}^{+} = \begin{pmatrix} 1/2 & 0 \\ 0 & 0 \end{pmatrix}$



The **pseudoinverse** of A, denoted A^+, is the unique linear operator $\mathbb{R}^m \to \mathbb{R}^n$ with

↑ plus sign; not to be confused with adjoint (†) or transpose (T)

- $R(A^+) = R(A^T)$ ✓
- $R(A^{+T}) = R(A)$ ✓
- $A^+A = P_{R(A^T)}$ ✓
- $AA^+ = P_{R(A)}$

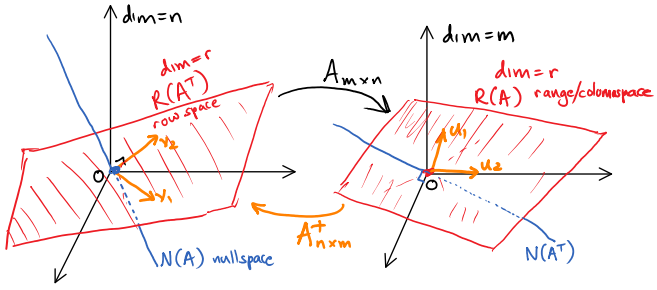
If the SVD of A is $A = \sum_j \sigma_j v_j u_j^T$ then A^+ is given by

$$A^+ = \sum_{j:\sigma_j>0} \frac{1}{\sigma_j} u_j v_j^T$$

$$A^+A = \sum_{j:\sigma_j>0} \frac{1}{\sigma_j} u_j v_j^T \sum_k \sigma_k v_k u_k^T = \sum_{j:\sigma_j>0} \frac{1}{\sigma_j}\sigma_j u_j u_j^T = \sum_{j:\sigma_j>0} u_j u_j^T = P_{R(A^T)}$$

$$AA^+ = \sum_{j:\sigma_j>0} v_j v_j^T = P_{R(A)}$$

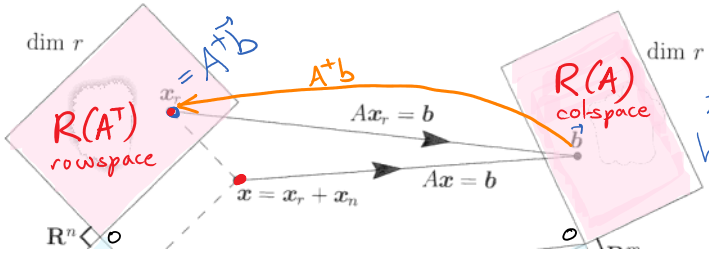
$\Rightarrow$ (for example) $A(A^+A) = A P_{R(A^T)} = A$ ✓ $\quad = P_{R(A)} A = A$ ✓



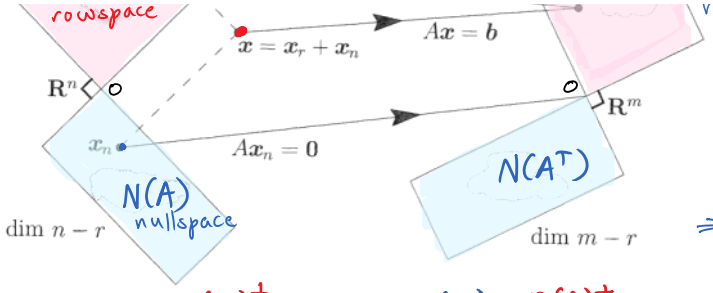
- If A is invertible/nonsingular, then $A^+ = A^{-1}$.

Corollary: If $A\vec{x} = \vec{b}$ has multiple solutions, ie., $N(A) \neq \{\vec{0}\}$, then $\vec{x} = A^+\vec{b}$ is the shortest solution.

Why? $A(A^+\vec{b}) = P_{R(A)}\vec{b} = \vec{b}$



Any solution $\vec{x}$ has the form $\vec{x} = A^+\vec{b} + \vec{n}$ where $\vec{n} \in N(A) \perp R(A^T)$



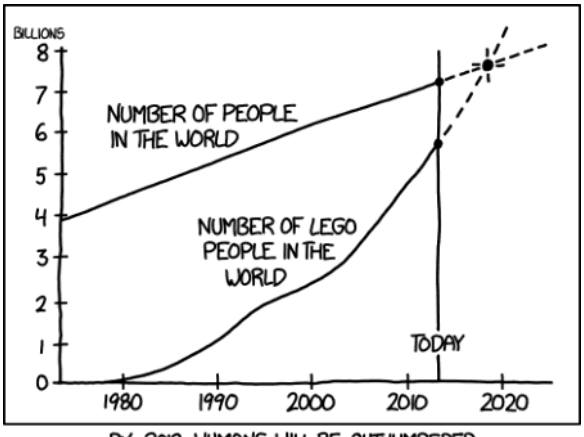
$N(A) = R(A^T)^\perp$ $\quad$ $N(A^T) = R(A)^\perp$

$\vec{x} = A^+\vec{b} + \vec{n}$
where $\vec{n} \in N(A) \perp R(A^T)$
$\|\vec{x}\|^2 = \|A^+\vec{b}\|^2 + \|\vec{n}\|^2$
$\Rightarrow$ shortest solution has $\vec{n} = \vec{0}$,
ie. $\vec{x} = A^+\vec{b}$. □

Why?
Any solution $x \in A^+b + N(A)$
(particular solution; set of solutions to the homogeneous equations $Ax = 0$)
Since $A^+b \in R(A^T)$, which is $\perp$ to $N(A)$,
$\|x\|^2 = \|A^+b\|^2 + \|\text{its component in } N(A)\|^2$
which is minimal if the component in $N(A)$ is 0,
ie., $x = A^+b$. ✓ □

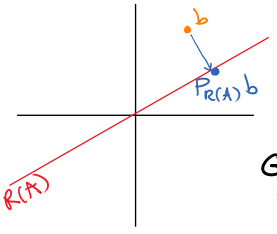
"Least Squares" Fitting & Applications to Data Analysis

(Reading: Meyer 4.6, 5.13-14; Strang 3.3)



BY 2019, HUMANS WILL BE OUTNUMBERED
[xkcd.com/128]

Typical problem: System of equations
$A\vec{x} = \vec{b}$
but $\vec{b} \notin R(A)$!
$\Rightarrow$ there is no solution $\vec{x}$
$\Rightarrow$ Instead, find x to minimize $\|Ax - b\|$



Geometrically:
Project b into $R(A)$,
solve
$Ax = P_{R(A)}b$

Example:

$$\underbrace{\begin{pmatrix} 1 & -1 \\ 2 & 3 \\ 0 & 0 \end{pmatrix}}_{A} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \underbrace{\begin{pmatrix} 0 \\ 5 \\ 10 \end{pmatrix}}_{b}$$

$$\begin{pmatrix} 2 & 3 \\ 0 & 0 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 5 \\ 10 \end{pmatrix}$$

A $\qquad$ b

There is no solution, since $(Ax)_3 = 0$ always.

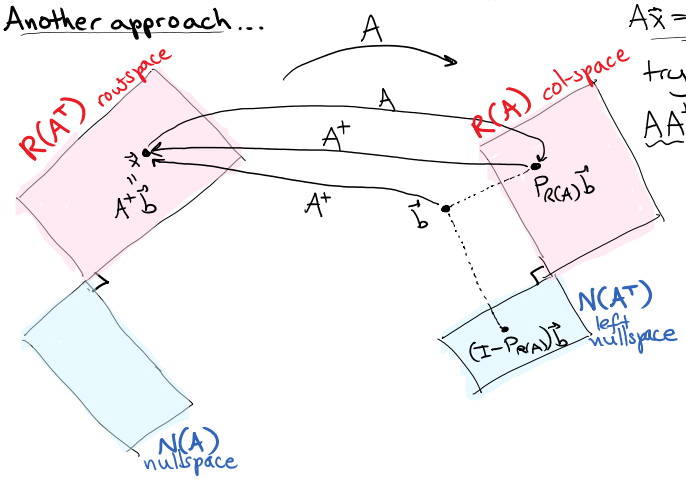
But $R(A) = \text{Span}\left\{\begin{pmatrix}1\\2\\0\end{pmatrix}, \begin{pmatrix}-1\\3\\0\end{pmatrix}\right\}$

$= xy$-plane in $\mathbb{R}^3$

$\hat{x} = A^+\vec{b}$ achieves $\min_x \|A\vec{x} - \vec{b}\|$ ✓

$A\vec{x} \overset{?}{=} \vec{b}$

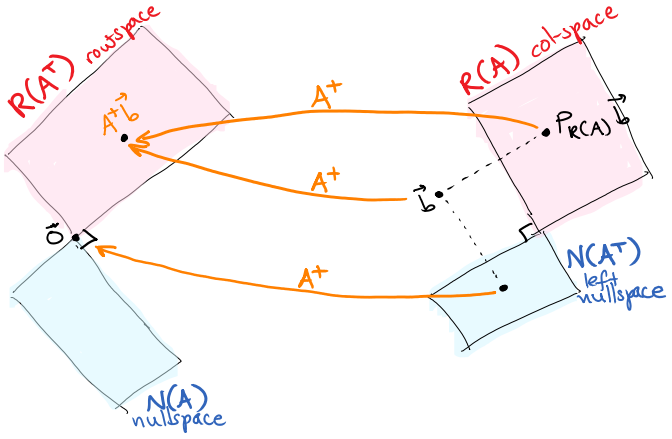
Another approach...



$A\hat{x} = P_{R(A)}\vec{b}$

try $x = A^+b$

$AA^+b = P_{R(A)}\vec{b}$ ✓



$\Rightarrow x = A^+b$ minimizes $\|Ax - b\|$

Algebraically:

$A = \sum_i \sigma_i \vec{u}_i \vec{v}_i$ (SVD)

$R(A) = \text{Span}(\{\vec{u}_i \mid \sigma_i > 0\})$

$P_{R(A)} = \sum_{i:\sigma_i>0} \vec{u}_i \vec{u}_i^T$ (since they are orthonormal)

We want to solve

$A\vec{x} = P_{R(A)}\vec{b}$

$$\sum_{i:\sigma_i>0} \sigma_i (v_i \cdot x)\vec{u}_i \qquad \sum_{i:\sigma_i>0} (u_i \cdot b)\vec{u}_i$$

$$\Rightarrow v_i \cdot x = \begin{cases} \frac{1}{\sigma_i}(u_i \cdot b) & \text{if } \sigma_i > 0 \\ 0 & \text{if } \sigma_i = 0 \end{cases}$$

$$\Rightarrow \vec{x} = \sum_i (v_i \cdot x)\vec{v}_i$$
$$= \sum_{i:\sigma_i>0} \frac{1}{\sigma_i} \vec{v}_i (u_i \cdot b)$$
$$= A^+ \vec{b}$$

$\Rightarrow x = A^+ b$ minimizes $\|Ax - b\|$

Example:

```
>> A = [1 -1; 2 3; 0 0]

A =

     1    -1
     2     3
     0     0

>> format long e
>> [U,D,V] = svd(A)
```

left singular vectors

```
U =

  -8.980559531591714e-02   9.959593139531120e-01                   0
   9.959593139531121e-01   8.980559531591698e-02                   0
                        0                        0   1.000000000000000e+00


D =

   3.618033988749895e+00                        0
                       0   1.381966011250106e+00
                       0                        0
```

singular values

right singular vectors

```
V =

   5.257311121191336e-01   8.506508083520399e-01
   8.506508083520399e-01  -5.257311121191336e-01

>> Apseudoinv = V * [1/3.618033988749895 0 0; 0 1/1.381966011250106 0] * U'

Apseudoinv =

   5.999999999999995e-01   1.999999999999999e-01                   0
  -3.999999999999997e-01   2.000000000000000e-01                   0

>> Apseudoinv * [0; 5; 10]

ans =

   9.999999999999996e-01
   1.000000000000000e+00
```

$= \begin{pmatrix} .6 & .2 & 0 \\ -.4 & .2 & 0 \end{pmatrix}$

$= \frac{1}{5}\begin{pmatrix} 3 & 1 & 0 \\ -2 & 1 & 0 \end{pmatrix}$

$= \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ ✓

```
>> format short e
>> Apseudoinv2 = pinv(A)
```

Matlab's built-in pseudo-inverse function

```
Apseudoinv2 =

   6.0000e-01   2.0000e-01            0
  -4.0000e-01   2.0000e-01            0
```

same as we got above!

```
>> A * Apseudoinv

ans =

   1.0000e+00  -1.3878e-16            0
            0   1.0000e+00            0
            0            0            0
```

Observe:

AA^+ = projection onto xy-plane

```
>> Apseudoinv * A

ans =
```

$A^+A = I$ on $\mathbb{R}^2$

```
>> format short e
>> Apseudoinv2 = pinv(A)

Apseudoinv2 =

   6.0000e-01   2.0000e-01            0
  -4.0000e-01   2.0000e-01            0
```

Matlab's built-in pseudo-inverse function

← same as we got above!

```
>> A * Apseudoinv

ans =

   1.0000e+00  -1.3878e-16            0
            0   1.0000e+00            0
            0            0            0
```

Observe: AA^+ = projection onto xy-plane

```
>> Apseudoinv * A

ans =

   1.0000e+00   1.1102e-16
   3.8858e-16   1.0000e+00
```

$A^+A = I$ on $\mathbb{R}^2$

APPLICATION : LINEAR REGRESSION

Example: Find the equation for the line $y = a + bx$ that goes through the points $(2, 5)$ and $(4, 11)$.

Answer: $a + 2\cdot b = 5$, $a + 4\cdot b = 11$ $\Leftrightarrow \begin{pmatrix} 1 & 2 \\ 1 & 4 \end{pmatrix}\begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} 5 \\ 11 \end{pmatrix}$

```
>> [1 2; 1 4] \ [5; 11]

ans =

   -1
    3
```

$\Rightarrow \boxed{y = -1 + 3x}$

Observe: $\begin{pmatrix} 1 & 2 \\ 1 & 4 \end{pmatrix}\begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} 5 \\ 11 \end{pmatrix}$

all 1's to account for the affine shift a; x-values; y-values

Example: Find the equation for the line $y = a + bx$ that goes through the points

$(2,5)$ —— $a + 2b = 5$

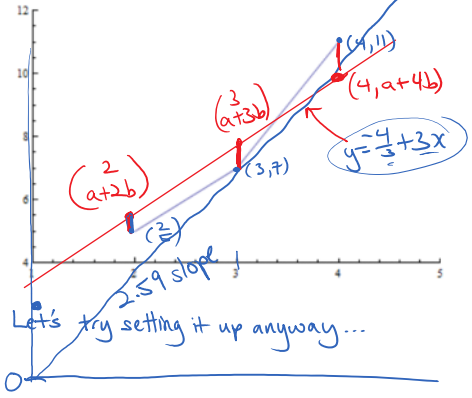
$(3,7)$ —— $a + 3b = 7$ $\longleftrightarrow \begin{pmatrix} 1 & 2 \\ 1 & 3 \\ 1 & 4 \end{pmatrix}\begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} 5 \\ 7 \\ 11 \end{pmatrix}$, with $\vec{b} = \begin{pmatrix} 5 \\ 7 \\ 11 \end{pmatrix}$

$(4,11)$ —— $a + 4b = 11$

Answer: There is no such line!

$A^+\vec{b} = \begin{pmatrix} -4/3 \\ 3 \end{pmatrix} = \vec{x}$

```
ListPlot[{{2, 5}, {3, 7}, {4, 11}},
 PlotRange → {{1, 5}, {4, 12}}, Joined → True]
```



This value for $\vec{x}$ minimizes $\|A\vec{x} - \vec{b}\|^2 = (a+2b-5)^2 + (a+3b-7)^2 + (a+4b-11)^2$

$= \sum_{\text{data points } j} (a + x_j b - y_j)^2$

Let's try setting it up anyway...

$y = bx$

$\begin{pmatrix} 2 \\ 3 \\ 4 \end{pmatrix}(b) = \begin{pmatrix} 5 \\ 7 \\ 11 \end{pmatrix}$, A, y

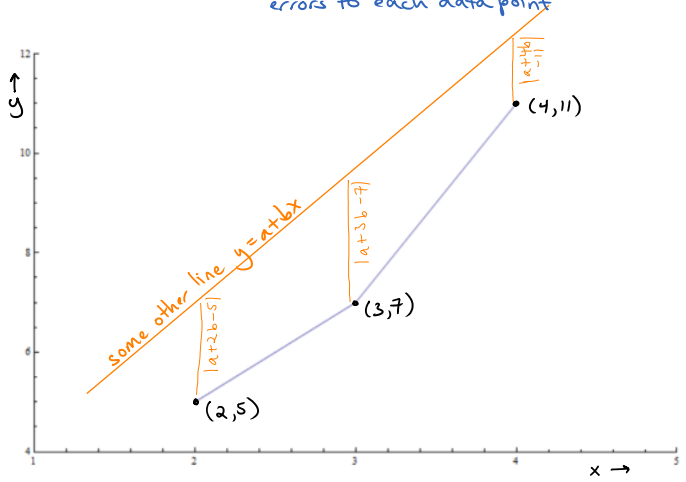
$b = 2.59$

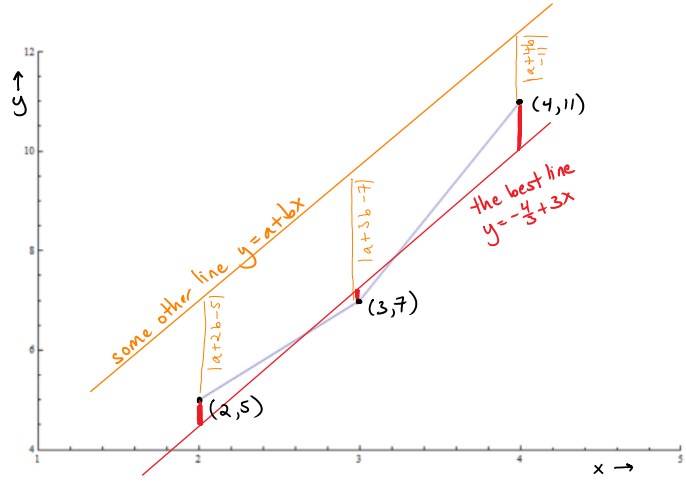
```
>> pinv([1 2; 1 3; 1 4]) * [5;7;11]

ans =

   -1.3333
    3.0000
```

$\Rightarrow (a,b) = (-\frac{4}{3}, 3)$ minimizes $\left\| \begin{pmatrix} 1 & 2 \\ 1 & 3 \\ 1 & 4 \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} - \begin{pmatrix} 5 \\ 7 \\ 11 \end{pmatrix} \right\|^2$

$= |a+2b-5|^2 + |a+3b-7|^2 + |a+4b-11|^2$

the sum of the squared errors to each data point





Example:
Predict what percent of Americans 16 or older will be employed in 2050.

Answer:
① Get the data.

Google percent us population working

Web News Images Shopping Videos More Search tools

About 247,000,000 results (0.47 seconds)

According to the October jobs report, the seasonally adjusted employment-to-population ratio was **59.2%** last month, one percentage point higher than it was a year earlier. Over that same period, the "official" unemployment rate fell from a seasonally adjusted **7.2%** to 5.8%. Nov 7, 2014

Employment, unemployment and underemployment ...
www.pewresearch.org/.../employment-vs-unemployme... Pew Research Center

Feedback

Employment-Population Ratio - Bureau of Labor Statistics ...
data.bls.gov/timeseries/LNS12300000 Bureau of Labor Statistics
Follow Us Follow BLS on Twitter | What's New | Release .... Labor Force Statistics from the Current Population Survey ... Type of data: **Percent** or rate. Age: 16 ...

Table A-1. Employment status of the civilian population by ...
www.bls.gov/news.release/empsit.t01.htm Bureau of Labor Statistics
Follow Us · Follow BLS on Twitter | What's New | Release Calendar ... Table A-1. Employment status of the civilian **population** by sex and age ... **Employed**. 146,941, 149,228, 148,980, 146,607, 148,795, 148,739, 148,840, 149,036, 148,800.

Employment-to-population ratio - Wikipedia, the free ...
https://en.wikipedia.org/wiki/Employment-to-**population**_ratio Wikipedia
Employment-to-population ratio in the world[edit] In general, a high ratio is considered to be above **70 percent** of the working-age population whereas a ratio below **50 percent** is considered to be low.

UNITED STATES DEPARTMENT OF LABOR

BUREAU OF LABOR STATISTICS

Home Subjects Data Tools Publications Economic Releases Students Beta

Databases, Tables & Calculators by Subject

Change Output Options: From: 2005 To: 2015 GO

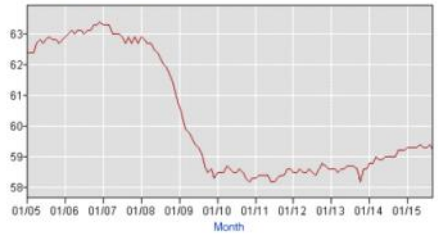
include graphs include annual averages

1948

Data extracted on: October 27, 2015 (10:44:11 AM)

Labor Force Statistics from the Current Population Survey

Series Id: LNS12300000
Seasonally Adjusted
Series title: (Seas) Employment-Population Ratio
Labor force status: Employment-population ratio
Type of data: Percent or rate
Age: 16 years and over



Download: xlsx

Year	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
2005	62.4	62.4	62.4	62.7	62.8	62.7	62.8	62.9	62.8	62.8	62.7	62.8
2006	62.9	63.0	63.1	63.0	63.1	63.1	63.0	63.1	63.1	63.3	63.3	63.4
2007	63.3	63.3	63.3	63.0	63.0	63.0	62.9	62.7	62.9	62.7	62.9	62.7
2008	62.9	62.8	62.7	62.7	62.5	62.4	62.2	62.0	61.9	61.7	61.4	61.0
2009	60.6	60.3	59.9	59.8	59.6	59.4	59.3	59.1	58.7	58.5	58.6	58.3
2010	58.5	58.5	58.5	58.7	58.6	58.5	58.5	58.6	58.5	58.3	58.2	58.3
2011	58.3	58.4	58.4	58.4	58.4	58.2	58.2	58.3	58.4	58.4	58.6	58.6
2012	58.5	58.5	58.6	58.5	58.5	58.6	58.5	58.4	58.6	58.8	58.7	58.6

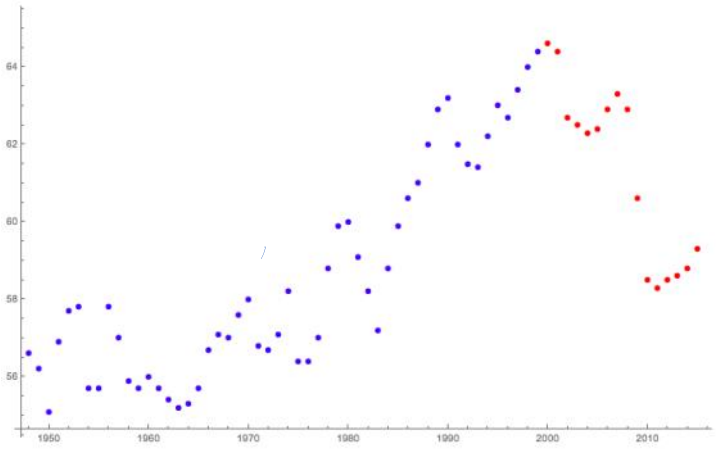
② Load the data into Mathematica / Matlab, and set up the matrices

```
data = Import["employment-population.csv"]
plot1 = ListPlot[data[[ ;; -17]], PlotMarkers → {Automatic, Small}, PlotStyle → Blue];
plot2 = ListPlot[data[[-16 ;;]], PlotMarkers → {Automatic, Small}, PlotStyle → Red];
plot = Show[plot1, plot2, PlotRange → {{1948, 2015}, {55, 65}}]
```


{{1948, 56.6}, {1949, 56.2}, {1950, 55.1}, {1951, 56.9}, {1952, 57.7}, {1953, 57.8}, {1954, 55.7}, {1955, 55.7}, {1956, 57.8}, {1957, 57.
{1958, 55.9}, {1959, 55.7}, {1960, 56.}, {1961, 55.7}, {1962, 55.4}, {1963, 55.2}, {1964, 55.3}, {1965, 55.7}, {1966, 56.7}, {1967, 57.1
{1968, 57.}, {1969, 57.6}, {1970, 58.}, {1971, 56.8}, {1972, 56.7}, {1973, 57.1}, {1974, 58.2}, {1975, 56.4}, {1976, 56.4}, {1977, 57.},
{1978, 58.8}, {1979, 59.9}, {1980, 60.}, {1981, 59.1}, {1982, 58.2}, {1983, 57.2}, {1984, 58.8}, {1985, 59.9}, {1986, 60.6}, {1987, 61.}
{1988, 62.}, {1989, 62.9}, {1990, 63.2}, {1991, 62.}, {1992, 61.5}, {1993, 61.4}, {1994, 62.2}, {1995, 63.}, {1996, 62.7}, {1997, 63.4},
{1998, 64.}, {1999, 64.4}, {2000, 64.6}, {2001, 64.4}, {2002, 62.7}, {2003, 62.5}, {2004, 62.3}, {2005, 62.4}, {2006, 62.9},
{2007, 63.3}, {2008, 62.9}, {2009, 60.6}, {2010, 58.5}, {2011, 58.3}, {2012, 58.5}, {2013, 58.6}, {2014, 58.8}, {2015, 59.3}}

```
data = Import["employment-population.csv"]
plot1 = ListPlot[data[[;; -17]], PlotMarkers → {Automatic, Small}, PlotStyle → Blue];
plot2 = ListPlot[data[[-16 ;;]], PlotMarkers → {Automatic, Small}, PlotStyle → Red];
plot = Show[plot1, plot2, PlotRange → {{1948, 2015}, {55, 65}}]
```

```
{{1948, 56.6}, {1949, 56.2}, {1950, 55.1}, {1951, 56.9}, {1952, 57.7}, {1953, 57.8}, {1954, 55.7}, {1955, 55.7}, {1956, 57.8}, {1957, 57.
 {1958, 55.9}, {1959, 55.7}, {1960, 56.}, {1961, 55.7}, {1962, 55.4}, {1963, 55.2}, {1964, 55.3}, {1965, 55.7}, {1966, 56.7}, {1967, 57.1
 {1968, 57.}, {1969, 57.6}, {1970, 58.}, {1971, 56.8}, {1972, 56.7}, {1973, 57.1}, {1974, 58.2}, {1975, 56.4}, {1976, 56.4}, {1977, 57.},
 {1978, 58.8}, {1979, 59.9}, {1980, 60.}, {1981, 59.1}, {1982, 58.2}, {1983, 57.2}, {1984, 58.8}, {1985, 59.9}, {1986, 60.6}, {1987, 61.}
 {1988, 62.}, {1989, 62.9}, {1990, 63.2}, {1991, 62.}, {1992, 61.5}, {1993, 61.4}, {1994, 62.2}, {1995, 63.}, {1996, 62.7}, {1997, 63.4},
 {1998, 64.}, {1999, 64.4}, {2000, 64.6}, {2001, 64.4}, {2002, 62.7}, {2003, 62.5}, {2004, 62.3}, {2005, 62.4}, {2006, 62.9},
 {2007, 63.3}, {2008, 62.9}, {2009, 60.6}, {2010, 58.5}, {2011, 58.3}, {2012, 58.5}, {2013, 58.6}, {2014, 58.8}, {2015, 59.3}}
```




```
A = data[[-16 ;;, 1]];
A = {1, #} & /@ A
b = data[[-16 ;;, 2]]
```

← insert 1's in each row

```
{{1, 2000}, {1, 2001}, {1, 2002}, {1, 2003}, {1, 2004}, {1, 2005}, {1,|2006}, {1, 2007},
 {1, 2008}, {1, 2009}, {1, 2010}, {1, 2011}, {1, 2012}, {1, 2013}, {1, 2014}, {1, 2015}}

{64.6, 64.4, 62.7, 62.5, 62.3, 62.4, 62.9, 63.3, 62.9, 60.6, 58.5, 58.3, 58.5, 58.6, 58.8, 59.3}
```

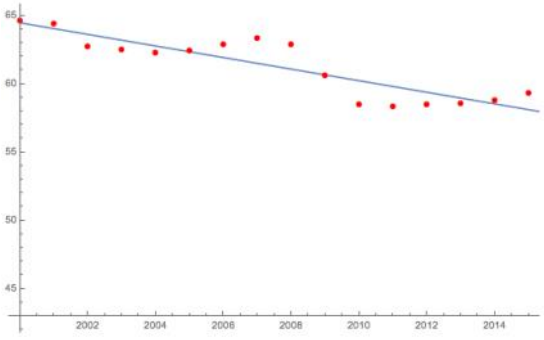
③ Compute the best-fitting line

```
{intercept, slope} = PseudoInverse[A].b
```

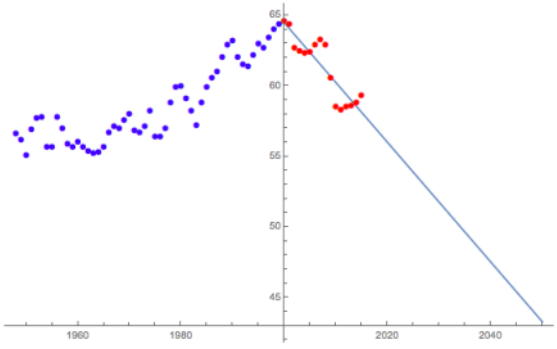
```
{912.113, -0.423824}
```

④ Evaluate it, and get the answer!

```
plot3 = Plot[intercept + slope x, {x, 2000, 2050}];
plot4 = Show[plot3, plot, PlotRange → {{2000, 2015}, Automatic}]
```




```
Show[plot4, PlotRange → {{1948, 2050}, Automatic}]
```



intercept + slope 2050

intercept + slope 1900

intercept + slope 2200

43.275 in 2050

106.849 in 1900

-20.2985 in 2200

Extrapolations often don't work well. ☹

LET'S GENERALIZE!

Setting: m data points

data pt. 1: $(x_1^{(1)}, x_2^{(1)}, x_3^{(1)}, \dots, x_k^{(1)}, y^{(1)})$

$(x_1^{(2)}, x_2^{(2)}, \dots, x_k^{(2)}, y^{(2)})$

$\vdots$

data pt. m: $(x_1^{(m)}, x_2^{(m)}, \dots, x_k^{(m)}, y^{(m)})$

components known exactly, eg., date/time

component that we want to predict

Goal: Find the best linear predictor for y,

$a_0, a_1, a_2, \dots, a_k \in \mathbb{R}$

to minimize total squared error.

$$\sum_{\text{data } j=1}^{m} \left| \underbrace{a_0 + a_1 x_1^{(j)} + \cdots + a_k x_k^{(j)}}_{\text{linear/affine prediction}} - \underbrace{y^{(j)}}_{\text{actual } j\text{th data}} \right|^2$$

$$\min \sum_{j=1}^{2} \left| a_0 + x_1^{(j)} a_1 + \cdots + x_k^{(j)} a_k - y^{(j)} \right|^2$$

Answer:

$$\begin{pmatrix} 1 & x_1^{(1)} & x_2^{(1)} & \cdots & x_k^{(1)} \\ 1 & x_1^{(2)} & x_2^{(2)} & \cdots & x_k^{(2)} \\ 1 & x_1^{(3)} & x_2^{(3)} & \cdots & x_k^{(3)} \\ & \vdots & & & \\ 1 & x_1^{(m)} & x_2^{(m)} & \cdots & x_k^{(m)} \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_k \end{pmatrix} = \begin{pmatrix} y^{(1)} \\ y^{(2)} \\ y^{(3)} \\ \vdots \\ y^{(m)} \end{pmatrix}$$

each row is one data point

A $\qquad$ $\vec{y}$

$$\sqrt{w} \begin{pmatrix} 1 & x_1^{(1)} & x_2^{(1)} & \cdots & x_k^{(1)} \\ 1 & x_1^{(2)} & x_2^{(2)} & \cdots & x_k^{(2)} \\ 1 & x_1^{(3)} & x_2^{(3)} & \cdots & x_k^{(3)} \\ & \vdots & & & \\ 1 & x_1^{(m)} & x_2^{(m)} & \cdots & x_k^{(m)} \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \\ \vdots \\ a_k \end{pmatrix} = \begin{pmatrix} \sqrt{w_1} y^{(1)} \\ \sqrt{w_2} y^{(2)} \\ \sqrt{w_3} y^{(3)} \\ \vdots \\ \sqrt{w_m} y^{(m)} \end{pmatrix}$$

A $\qquad$ $\vec{y}$

The least-squares solution is $A^+ \vec{y}$.

pseudoinverse

$\Rightarrow$ if you want to minimize

$$\sum_{\text{data } j} w_j \left| a_0 + a_1 x^{(j)} - y^{(j)} \right|^2$$

$\sqrt{W} A \begin{pmatrix} a_0 \\ a_1 \end{pmatrix} = \sqrt{W} \vec{y}$ where $W = \begin{pmatrix} w_1 & & 0 \\ & \ddots & \\ 0 & & w_m \end{pmatrix}$

$\Rightarrow$ solution is $(\sqrt{W} A)^+ \sqrt{W} \vec{y} \neq A^+ \vec{y}$ in general

$[(AB)^{-1} = B^{-1} A^{-1}$ if both are invertible$]$

Example:

Consider the time (T) it takes for a runner to complete a marathon (26 miles and 385 yards). Many factors such as height, weight, age, previous training, etc. can influence an athlete's performance, but experience has shown that the following three factors are particularly important:

$x_1 = \text{Ponderal index} = \dfrac{\text{height (in.)}}{[\text{weight (lbs.)}]^{\frac{1}{3}}}$, $= \frac{1}{\text{BMI}}$

$x_2 = \text{Miles run the previous 8 weeks},$

$x_3 = \text{Age (years)}.$

A linear model hypothesizes that the time T (in minutes) is given by $T = \alpha_0 + \alpha_1 x_1 + \alpha_2 x_2 + \alpha_3 x_3 + \varepsilon$, where ε is a random function accounting for all other factors and whose mean value is assumed to be zero. On the basis of the five observations given below, estimate the expected marathon time for a 43-year-old runner of height 74 in., weight 180 lbs., who has run 450 miles during the previous eight weeks.

Answer: ~3:50

	Time T in minutes	1/BMI x_1	miles x_2	age x_3	
runners	3:01 = 181	13.1	619	23	α_0
	193	13.5	803	42	$\alpha_1 < 0$
	212	13.8	207	31	$\alpha_2 < 0$
	221	13.1	409	38	
	248	12.5	482	45	$\alpha_3 > 0$

What is your personal predicted mean marathon time?

T $\quad \frac{74}{180^{1/3}}$ $\quad 450$ $\quad 43$

$$T = \alpha_0 + \alpha_1 x_1 + \alpha_2 x_2 + \alpha_3 x_3 + \alpha_4 x_2^2$$

Answer: Start with a sanity check: $\alpha_1 < 0$, $\alpha_2 < 0$, $\alpha_3 > 0$

① Enter the data ② Find best-fit plane ③ Predict!

```
>> A = [
1 13.1 619 23 ;
1 13.5 803 42 ;
1 13.8 207 31 ;
1 13.1 409 38 ;
1 12.5 482 45
]; x1 x2 x3
>> b = [
181; 193; 212; 221; 248
];
```

A: times in minutes

```
>> alpha = pinv(A) * b

alpha =

  492.0442
  -23.4355
   -0.0761
    1.8624
```

every year older = ~2 minutes slower

2 miles more/week = 1 min. faster

x1 x2 x3

```
>> alpha' * [1; 74/180^(1/3); 450; 43]

ans =

  230.7209
```

← predicted time = 3 hours 50 minutes

Example (continued): Same problem, but now fit a quadratic curve to x_2 (distance over previous 8 weeks) — running too much slows you down?

Answer: Want $T = \alpha_0 + \alpha_1 x_1 + \alpha_2 x_2 + \alpha_3 x_3 + \alpha_4 x_2^2$

① Enter the data & find best-fit hyperplane

```
>> A2 = [A, A(:,3).^2];
alpha2 = pinv(A2) * b
```

② Predict!

```
>> alpha2' * [1; 74/180^(1/3); 450; 43; 450^2]
```

① Enter the data & find best-fit hyperplane

```
>> A2 = [A, A(:,3).^2];
alpha2 = pinv(A2) * b

alpha2 =

  740.7697
  -38.7075
   -0.2441
    1.5801
    0.0002
```

② Predict!

```
>> alpha2' * [1; 74/180^(1/3); 450; 43; 450^2]

ans =

  224.3594
```

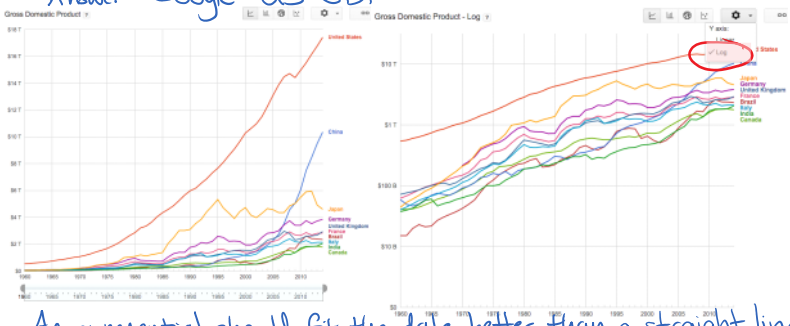
= 3 hours 44 minutes

(Of course, be careful of overfitting the data by using too many parameters.

Fitting other curves than lines:

Example: Predict US gross domestic product (GDP) in 2050.

Answer: Google "US GDP"



An exponential should fit the data better than a straight line

$G = e^{a+bT}$

$\Rightarrow \log G = a + bT$

years = [1966:2014];
GDP = [3800176644539.76, 3895181060653.26, 4082149751564.6, 4208696393863.11, 4343661175265.73, 4486805518455.5, 4722957883148.85, 4989480292992.24, 4963676968025.7, 4953864844037.75, 5220684465530.35, 5461284794411.99, 5765024247749.54, 5948103589527.53, 5933554752676.08, 6087499073703.77, 5971173597636.82, 6247785657762.76, 6701317851694.21, 6985369125864.67, 7230668360903.64, 7480975855992.56, 7795473984765.14, 8082388278264.36, 8237519238199.96, 8231416510730, 8524075976250.96, 8758134889170.93, 9111757146664.33, 9359503617414.74, 9714779258395.62, 10150683977472.6, 10602380376634.4, 11099123060521.4, 11553318760428.6, 11666077052746.7, 11874448085025.7, 12207737238841, 12669890778469.9, 13093726000000, 13442886679117.7, 13681977860942.7, 13642078277526.4, 13263438360402.5, 13599258090661.7, 13817044044776, 14137749306899.7, 14451509513869.6, 14796640462032];

```
>> A = [ones(length(years),1) years']

A =

     1     1966
     1     1967
     1     1968
     1     1969
     1     1970
     1     1971
     1     1972
```

$\begin{pmatrix} 1 & 1960 \\ 1 & 1961 \\ 1 & 1962 \\ \vdots & \vdots \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} \log(5.2\times10^{11}) \\ \log(5.4\times10^{11}) \\ \log(5.8\times10^{11}) \\ \vdots \end{pmatrix}$

```
>> b = log(GDP)'
>> x = pinv(A) * b

x =

  -29.0848
    0.0295
```

```
>> x = A \ b

x =

  -29.0848
    0.0295
```

```
>> exp([1 2050] * x)

ans =

  4.7132e+13
```

= $47 trillion

Also easy to fit other functions, eg., $a + bT + cT^2 = \log G$:

```
>> B = [A (years') .^2]

B =

     1     1966     3865156
     1     1967     3869089
     1     1968     3873024
     1     1969     3876961
     1     1970     3880900
     1     1971     3884841
     1     1972     3888784
```

```
>> y = B \ b

y =

  -588.2974
     0.5916
    -0.0001
```

```
>> exp([1 2050 2050^2] * y)

ans =

  2.9161e+13
```

Four ways to find x to minimize $\|Ax-b\|^2$:

① Compute $P_{R(A)}\vec{b}$ (eg., using Gram-Schmidt to get on. basis for R(A))

Solve $A\vec{x} = P_{R(A)}\vec{b}$

① Compute pseudoinverse A^+, set $\vec{x} = A^+\vec{b}$.

Pros: Easy to remember, one line pinv(A)*b in Matlab

Cons: Slow for large A, numerically unstable

(**Caution!** Generalized inverses are useful in formulating theoretical statements

Pros: Easy to remember, one line pinv(A)*b in Matlab
Cons: Slow for large A, numerically unstable

Meyer p.423: **Caution!** Generalized inverses are useful in formulating theoretical statements such as those above, but, just as in the case of the ordinary inverse, generalized inverses are not practical computational tools. In addition to being computationally inefficient, serious numerical problems result from the fact that A^+ need not be a continuous functions of the entries of A.

② Find x manually, with calculus:

$$E_{\text{squared error}} = \left\| \begin{pmatrix} 1 & a_1 \\ 1 & a_2 \\ 1 & a_3 \\ \vdots & \vdots \\ 1 & a_m \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix} \right\|^2$$

$$= \sum_{j=1}^{m} (x_1 + a_j x_2 - b_j)^2$$

$$\frac{\partial}{\partial x_1} = 2\sum_j (x_1 + a_j x_2 - b_j) = 0 \quad \Rightarrow m x_1 + \left(\sum_j a_j\right) x_2 = \left(\sum_j b_j\right)$$

$$\frac{\partial E}{\partial x_2} = 2\sum_j a_j (x_1 + a_j x_2 - b_j) = 0 \quad \Rightarrow \left(\sum_j a_j\right) x_1 + \left(\sum_j a_j^2\right) x_2 = \sum_j a_j b_j$$

Let $\bar{a} = \frac{1}{m}\sum_j a_j$ (average value of a_j's)
$\bar{b} = \frac{1}{m}\sum_j b_j$ (average of b_j's)

First equation
$\Rightarrow$ intercept $x_1 = \bar{b} - \bar{a} x_2$
(this is 0 if the data is centered so $\bar{a} = \bar{b} = 0$)

Second equation
$$\Rightarrow x_2 = \frac{\sum_j a_j b_j - m \cdot \bar{a}\bar{b}}{\sum_j a_j^2 - m(\bar{a})^2}$$

$$\left(= \frac{\text{Cov}(A,B)}{\text{Var}(A)} \quad \text{if } A \text{ and } B \text{ are random variables with } \Pr[(A,B) = (a_j, b_j)] = \frac{1}{m}. \right)$$

Pros: None (okay, it gives an easy closed-form solution)

③ Solve $A^T A x = A^T b$ exact!

Claim: This has the same solution, ie. $A^+ \vec{b}$.
in other words, in Matlab

$$\text{pinv}(A) * b = (A^T A) \backslash (A^T b)$$

Proof: Plug in $x = A^+ b$:

$$A^T (A A^+) \vec{b} = A^T P_{R(A)} \vec{b} = A^T b$$

($A A^+ = P_{R(A)}$; $P_{R(A)} A = A$; $(P_{R(A)} A)^T = A^T$; $A^T P_{R(A)}^T = A^T$; $P_{R(A)}^T = P_{R(A)}$) ☑

Theorem: Let A be an $m \times n$ real matrix with rank n.

- The equation $A^T A x = A^T b$ is feasible (ie, it has a solution x)
- The unique solution is

(i.e., it has a solution x)

- The unique solution is

$$(A^TA)^{-1}A^Tb = A^+b$$

(pseudoinverse)

Proof: There is a solution because
- $R(A^TA) = R(A^T)$ (shown above)
- $A^Tb \in R(A^T)$

$\Rightarrow A^Tb \in R(A^TA)$ ✓

The solution is unique because $N(A) = \{\vec{0}\}$
(because $\text{rank}(A) = \dim R(A^T) = n$, by assumption,
and by rank-nullity $\dim R(A^T) + \dim N(A) = n$)

So why is $(A^TA)^{-1}A^Tb = A^+b$, as claimed?
There are several ways of seeing it.

Algebraically: Using the SVD of A, $A = \sum_i \lambda_i \vec{u}_i \vec{v}_i^T$,

$$A^+ = \sum_{i:\lambda_i>0} \frac{1}{\lambda_i} v_i u_i^T$$

$$(A^TA)^{-1}A^T = \left[\sum_i \lambda_i^2 v_i v_i^T\right]^{-1}\left(\sum_j \lambda_j v_j u_j^T\right)$$

$$= \left(\sum_i \frac{1}{\lambda_i^2} v_i v_i^T\right)\left(\sum_j \lambda_j v_j u_j^T\right)$$

↳ since $\text{rank}(A) = n$, $\lambda_1, \ldots, \lambda_n$ are all > 0

$$= \sum_i \frac{1}{\lambda_i} v_i u_i^T$$

$$= A^+ \quad ✓$$

More geometrically:

$x = A^+b$ solves $Ax = P_{R(A)}b$

$\Rightarrow A^TAx = A^TP_{R(A)}b$

$= A^Tb$ since $A^T = A^TP_{R(A)}$

(in general $A = P_{R(A)}AP_{R(A^T)}$)

□

Corollary: For any $m \times n$ real matrix A, of rank n,

$$A^+ = (A^TA)^{-1}A^T$$

Proof: Since we just showed $A^+b = (A^TA)^{-1}A^Tb$ for any b. □

$$A^TA\vec{x} = A^T\vec{b}$$

Pros: Fast

Cons: condition number of A^TA = (condition number of A)^2

Method ④ Solve $\begin{pmatrix} I & A \\ A^T & 0 \end{pmatrix}\begin{pmatrix} y \\ x \end{pmatrix} = \begin{pmatrix} b \\ 0 \end{pmatrix}$

Pros: Maintains sparsity of A.

[Problem 4.6.9, p.237 of Meyer]

⑤ Latest research has focused on finding fast randomized approximation algorithms, based on dimension reduction à la Johnson-Lindenstrauss Lemma, eg.,

[Clarkson Woodruff 2012 http://arxiv.org/abs/1207.6365]

[Nelson & Nguyen 2012 http://arxiv.org/abs/1211.1002]

$A = \begin{pmatrix} 100 & 0 \\ 0 & 1 \end{pmatrix}$ $\quad \|A^{-1}\| = 1$

$\|A^{-1}\| = \begin{pmatrix} \frac{1}{100} & 0 \\ 0 & 1 \end{pmatrix}$

$B = \begin{pmatrix} 100 & 0 \\ 0 & \frac{1}{1000} \end{pmatrix}$ $\quad \|B^{-1}\| = 1000$

$\|B^{-1}\| = \begin{pmatrix} \frac{1}{100} & 0 \\ 0 & 1000 \end{pmatrix}$

A sing values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n > 0$

$\|A\| = \sigma_1$ $\quad \|A^{-1}\| = \frac{1}{\sigma_n}$

$F = \begin{pmatrix} 1 & 1 & -1 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{pmatrix}$ $\quad \sigma = \sqrt{3}$

$\vec{v} = e_3$ $\quad \vec{u} = \frac{1}{\sqrt{3}}\begin{pmatrix} -1 \\ 1 \\ 1 \end{pmatrix}$

$G = \begin{pmatrix} 1 & -2 & 0 \\ 0 & 1 & 1 \end{pmatrix}$

$\vec{u} = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ 0 \end{pmatrix}$ $\quad \vec{v} = \frac{Fu}{\|Fu\|}$ $\quad \sigma = \|Fu\|$

$u = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ 0 \end{pmatrix}$ $\quad \sigma = 1$ $\quad v = \begin{pmatrix} 0 \\ -1 \\ 1 \end{pmatrix}/\sqrt{2}$

$G^T = \begin{pmatrix} 1 & 0 \\ -2 & 1 \\ 0 & 1 \end{pmatrix}$ $\quad \begin{pmatrix} 1 \\ y \end{pmatrix}$ $\quad \frac{d}{dy}\frac{\|G^T(\frac{1}{y})\|^2}{\|(\frac{1}{y})\|^2} = 0$

$\left\| \begin{pmatrix} 1 & -1 \end{pmatrix} \right\|_{1\times 2} = \sqrt{2}$

$\left\| \begin{pmatrix} 1 & .01 \end{pmatrix} \right\| = \frac{1}{\sqrt{1^2+.01^2}}\begin{pmatrix} 1 \\ .01 \end{pmatrix}$

$\mathbb{E}_{X\sim D}\left[(f(X)-p(X))^2\right]$

$= \int dx\, q_D(x)\,(f(x)-p(x))^2$

$= \|f-p\|^2 = \langle f-p, f-p\rangle$

where $\langle a,b\rangle = \int dx\, q_D(x)\, a(x)\, b(x)$