Solving Linear System of Equations

The "Undo" button for Linear Operations

Matrix-vector multiplication: given the data $\boldsymbol{x}$ and the operator $\boldsymbol{A}$, we can find $\boldsymbol{y}$ such that

$$\boldsymbol{y} = \boldsymbol{A}\,\boldsymbol{x}$$

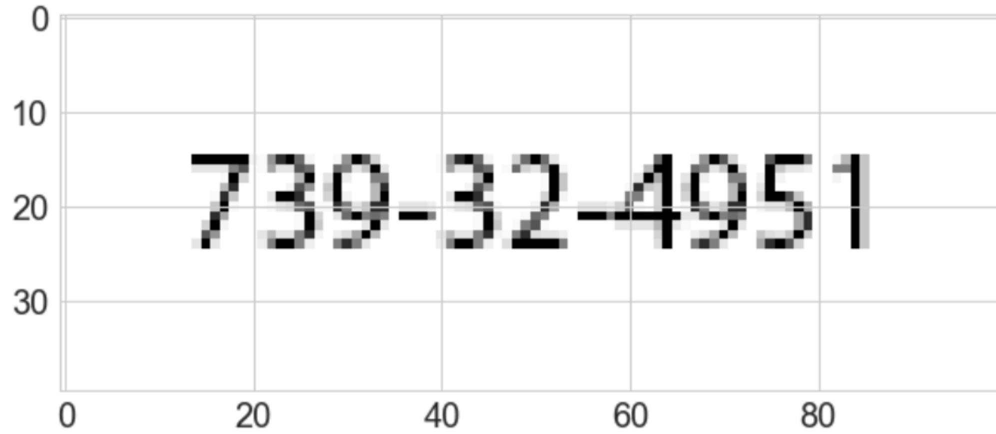
$$\boldsymbol{x} \xrightarrow[\text{transformation}]{\boldsymbol{A}} \boldsymbol{y}$$

What if we know $\boldsymbol{y}$ but not $\boldsymbol{x}$? How can we "undo" the transformation?

$$\boldsymbol{y} \xrightarrow[?]{\boldsymbol{A}^{-1}} \boldsymbol{x}$$

Solve $\boldsymbol{A}\,\boldsymbol{x} = \boldsymbol{y}$ for $\boldsymbol{x}$

Image Blurring Example



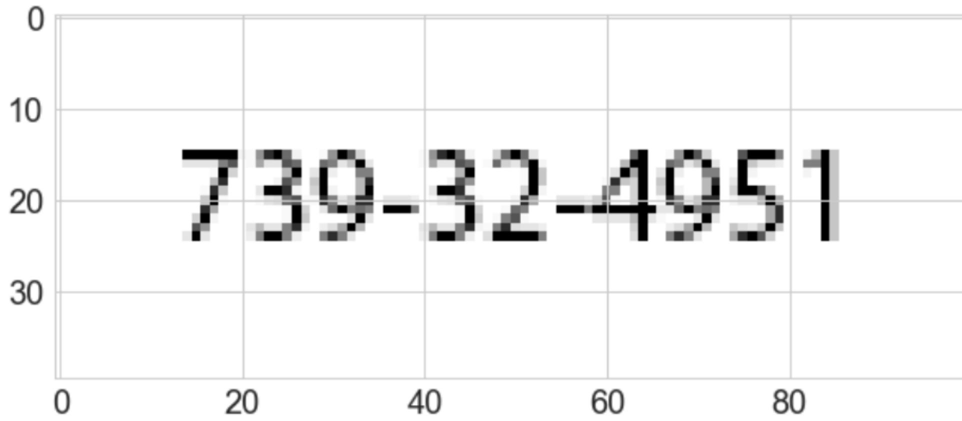
- Image is stored as a 2D array of real numbers between 0 and 1
  (0 represents a white pixel, 1 represents a black pixel)
- $\boldsymbol{xmat}$ has 40 rows of pixels and 100 columns of pixels
- Flatten the 2D array as a 1D array
- $\boldsymbol{x}$ contains the 1D data with dimension 4000,
- Apply blurring operation to data $\boldsymbol{x}$, i.e.

$$\boldsymbol{b} = \boldsymbol{A}\,\boldsymbol{x}$$

where $\boldsymbol{A}$ is the blur operator and $\boldsymbol{b}$ is the blurred image

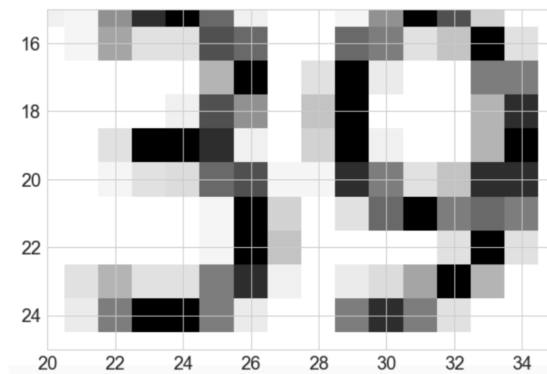
Blur operator

$\boldsymbol{x}$




```
plt.imshow(xmat);
plt.axis([20,35,25,15])
```

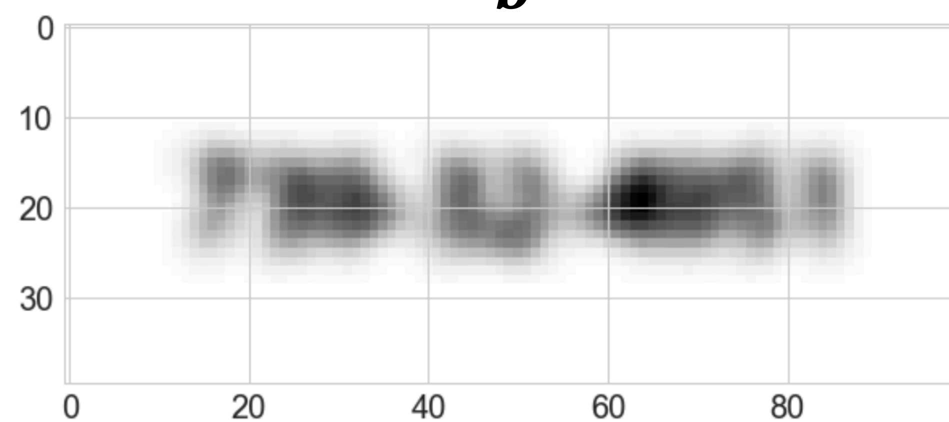
[20, 35, 25, 15]



Blur operator

$\boldsymbol{A}$

$\boldsymbol{b}$



blurred image (4000,)

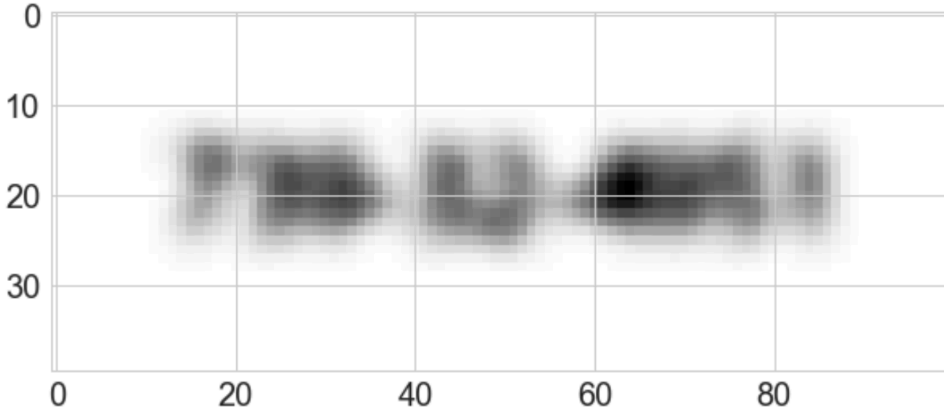
Blur operator (4000,4000)

"original" image (4000,)

$$\boldsymbol{b} = \boldsymbol{A}\,\boldsymbol{x}$$

"Undo" Blur to recover original image

$\boldsymbol{b}$

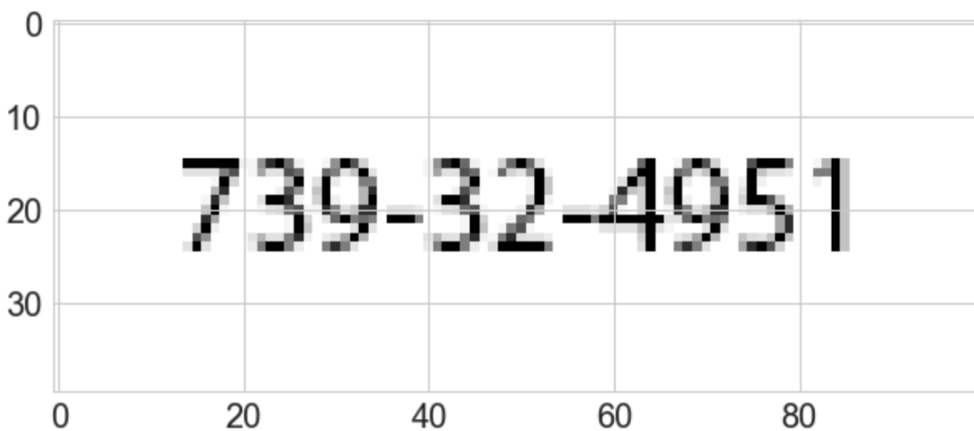


Solve

$\boldsymbol{A}\,\boldsymbol{x} = \boldsymbol{b}$

for $\boldsymbol{x}$

$\boldsymbol{x}$



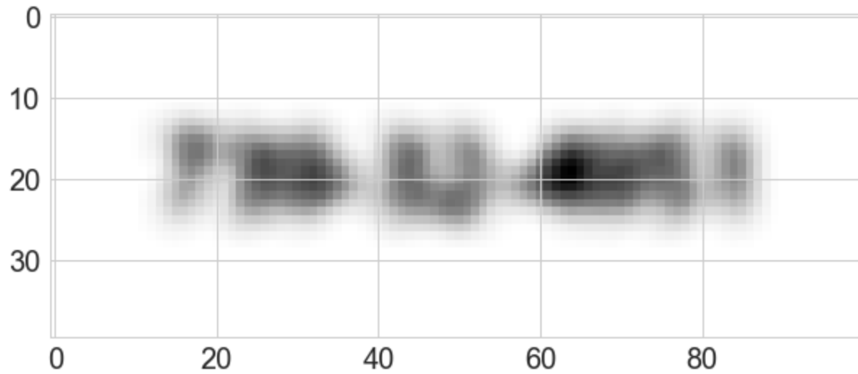
Assumptions:

1. we know the blur operator $\boldsymbol{A}$
2. the data set $\boldsymbol{b}$ does not have any noise ("clean data"

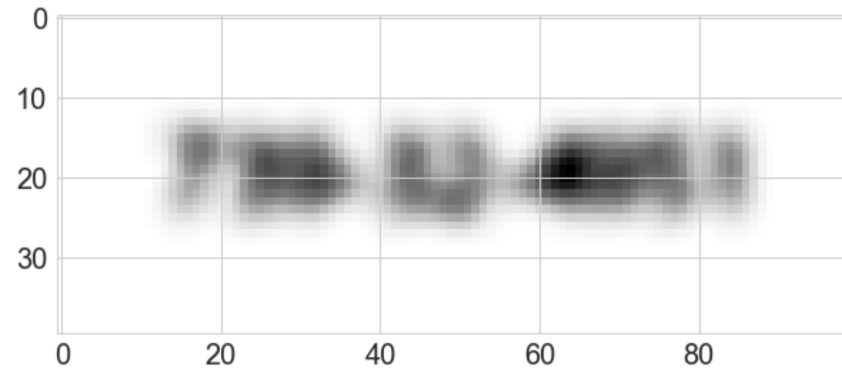
What happens if we add some noise to $\boldsymbol{b}$?

”Undo” Blur to recover original image

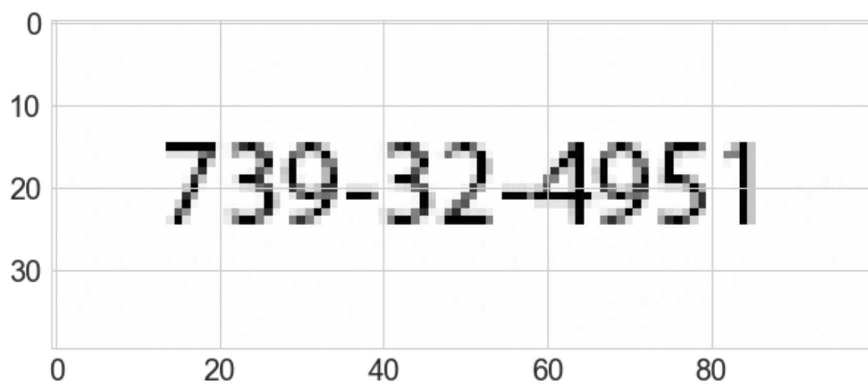
$\boldsymbol{y} + a * 10^{-6}\ (a \in (0,1))$

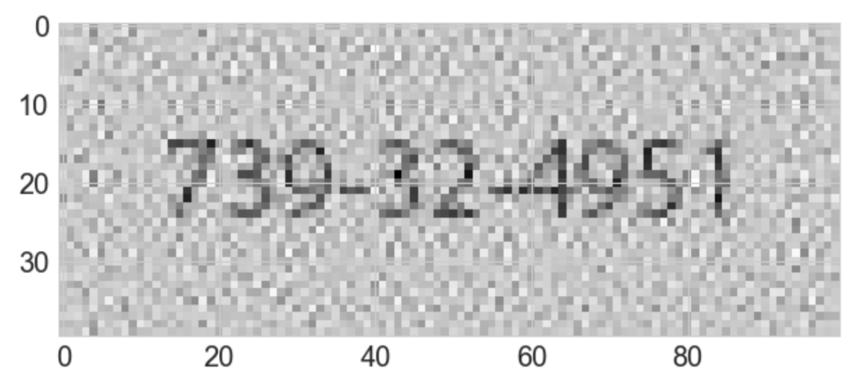


$\boldsymbol{y} + a * 10^{-4}\ (a \in (0,1))$



Solve $\boldsymbol{A}\,\boldsymbol{x} = \boldsymbol{y}$ for $\boldsymbol{x}$





How much noise can we add and still be able to recover meaningful information from the original image? At which point this inverse transformation fails?

We will talk about sensitivity of the “undo” operation later.

Linear System of Equations

How do we actually solve $\boldsymbol{A}\,\boldsymbol{x} = \boldsymbol{b}$?

We can start with an "easier" system of equations…

Let's consider triangular matrices (lower and upper):

$$\begin{pmatrix} L_{11} & 0 & \dots & 0 \\ L_{21} & L_{22} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ L_{n1} & L_{n2} & \dots & L_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b \\ b_2 \\ \vdots \\ b_n \end{pmatrix}$$

$$\begin{pmatrix} U_{11} & U_{12} & \dots & U_{1n} \\ 0 & U_{22} & \dots & U_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & U_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} b \\ b_2 \\ \vdots \\ b_n \end{pmatrix}$$

Example: Forward-substitution for lower triangular systems

$$\begin{pmatrix} 2 & 0 & 0 & 0 \\ 3 & 2 & 0 & 0 \\ 1 & 2 & 6 & 0 \\ 1 & 3 & 4 & 2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 2 \\ 2 \\ 6 \\ 4 \end{pmatrix}$$

$$2\,x_1 = 2 \rightarrow x_1 = 1$$

$$3\,x_1 + 2\,x_2 = 2 \rightarrow x_2 = \frac{2-3}{2} = -0.5$$

$$1\,x_1 + 2\,x_2 + 6\,x_3 = 6 \rightarrow x_3 = \frac{6-1+1}{6} = 1.0$$

$$1\,x_1 + 3\,x_2 + 4\,x_3 + 2\,x_4 = 4 \rightarrow x_3 = \frac{4-1+1.5-4}{2} = 0.25$$

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 1 \\ -0.5 \\ 1.0 \\ 0.25 \end{pmatrix}$$

Example: Backward-substitution for upper triangular systems

$$\begin{pmatrix} 2 & 8 & 4 & 2 \\ 0 & 4 & 4 & 3 \\ 0 & 0 & 6 & 2 \\ 0 & 0 & 0 & 2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 2 \\ 4 \\ 4 \\ 1 \end{pmatrix}$$

$$x_4 = \frac{1}{2}$$

$$x_3 = \frac{4 - 2\,\frac{1}{2}}{6} = \frac{1}{2}$$

$$x_2 = \frac{4 - 4\,\frac{1}{2} - 3\,\frac{1}{2}}{4} = \frac{1/2}{4} = \frac{1}{8}$$

$$x_1 = \frac{2 - 8\,\frac{1}{8} - 4\,\frac{1}{2} - 2\,\frac{1}{2}}{2} = \frac{-2}{2} = -1$$

LU Factorization

How do we solve $\boldsymbol{A}\,\boldsymbol{x} = \boldsymbol{b}$ when $\boldsymbol{A}$ is a non-triangular matrix?

We can perform LU factorization: given a $n \times n$ matrix $\boldsymbol{A}$, obtain lower triangular matrix $\boldsymbol{L}$ and upper triangular matrix $\boldsymbol{U}$ such that

$$\boldsymbol{A} = \boldsymbol{L}\boldsymbol{U}$$

where we set the diagonal entries of $\boldsymbol{L}$ to be equal to 1.

$$\begin{pmatrix} 1 & 0 & \dots & 0 \\ L_{21} & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ L_{n1} & L_{n2} & \dots & 1 \end{pmatrix} \begin{pmatrix} U_{11} & U_{12} & \dots & U_{1n} \\ 0 & U_{22} & \dots & U_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & U_{nn} \end{pmatrix} = \begin{pmatrix} A_{11} & A_{12} & \dots & A_{1n} \\ A_{21} & A_{22} & \dots & A_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n1} & A_{n2} & \dots & A_{nn} \end{pmatrix}$$

LU Factorization

$$\begin{pmatrix} 1 & 0 & \dots & 0 \\ L_{21} & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ L_{n1} & L_{n2} & \dots & 1 \end{pmatrix} \begin{pmatrix} U_{11} & U_{12} & \dots & U_{1n} \\ 0 & U_{22} & \dots & U_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & U_{nn} \end{pmatrix} = \begin{pmatrix} A_{11} & A_{12} & \dots & A_{1n} \\ A_{21} & A_{22} & \dots & A_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n1} & A_{n2} & \dots & A_{nn} \end{pmatrix}$$

Assuming the LU factorization is know, we can solve the general system

LU Factorization (with pivoting)

Factorize: $\boldsymbol{A} = \boldsymbol{PLU}$

$$\boldsymbol{PL}\underbrace{\boldsymbol{U\,x}}_{\boldsymbol{y}} = \boldsymbol{b}$$

Forward-substitution $\boldsymbol{L\,y} = \boldsymbol{P}^{\boldsymbol{T}}\boldsymbol{b}$ (Solve for $\boldsymbol{y}$)

Backward-substitution $\boldsymbol{U\,x} = \boldsymbol{y}$ (Solve for $\boldsymbol{x}$)

Example

Assume the $\boldsymbol{A} = \boldsymbol{LU}$ factorization is known, yielding:

$$\boldsymbol{L} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0.5 & 1 & 0 & 0 \\ 0.5 & 1 & 1 & 0 \\ 0.5 & 0.5 & 0.5 & 1 \end{pmatrix} \quad \boldsymbol{U} = \begin{pmatrix} 2 & 8 & 4 & 1 \\ 0 & -2 & 1 & 2.5 \\ 0 & 0 & 3 & -1 \\ 0 & 0 & 0 & 0.75 \end{pmatrix}$$

Determine the solution $\boldsymbol{x}$ that satisfies $\boldsymbol{Ax} = \boldsymbol{b}$, when $\boldsymbol{b} = \begin{pmatrix} 2 \\ 2 \\ 1 \\ 4 \end{pmatrix}$

$$\boldsymbol{L}\,\underbrace{\boldsymbol{U}\boldsymbol{x}}_{\boldsymbol{y}} = \boldsymbol{b}$$

First, solve the lower-triangular system $\boldsymbol{L}\,\boldsymbol{y} = \boldsymbol{b}$ for the variable $\boldsymbol{y}$

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0.5 & 1 & 0 & 0 \\ 0.5 & 1 & 1 & 0 \\ 0.5 & 0.5 & 0.5 & 1 \end{pmatrix} \boldsymbol{y} = \begin{pmatrix} 2 \\ 2 \\ 1 \\ 4 \end{pmatrix}$$

Then, solve the upper-triangular system $\boldsymbol{U}\,\boldsymbol{x} = \boldsymbol{y}$ for the variable $\boldsymbol{x}$

$$\begin{pmatrix} 2 & 8 & 4 & 1 \\ 0 & -2 & 1 & 2.5 \\ 0 & 0 & 3 & -1 \\ 0 & 0 & 0 & 0.75 \end{pmatrix} \boldsymbol{x} = \begin{pmatrix} 2 \\ 1 \\ -1 \\ 3 \end{pmatrix}$$

Methods to solve linear system of equations

$$\boldsymbol{A}\,\boldsymbol{x} = \boldsymbol{b}$$

- LU
- Cholesky
- Sparse