CS395T
Computational Statistics with Application to Bioinformatics

Prof. William H. Press
Spring Term, 2011
The University of Texas at Austin

Lecture 28

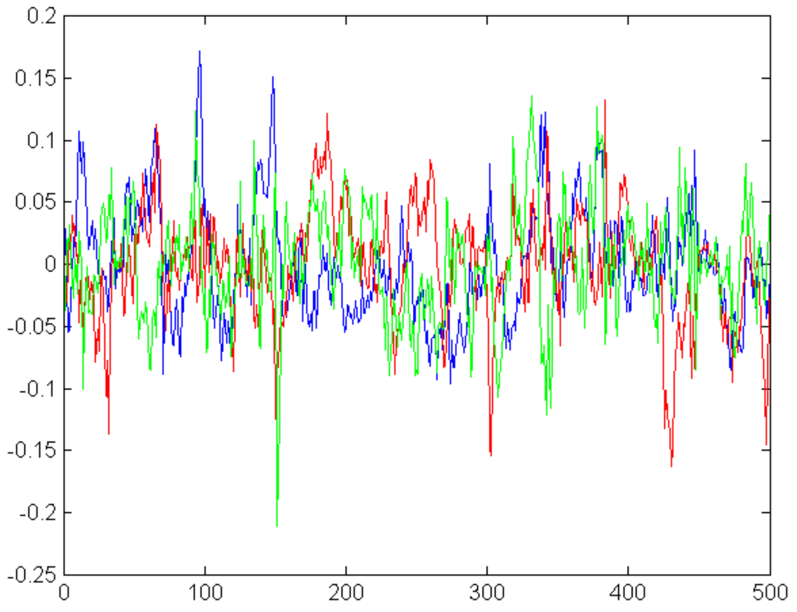
Eigengenes and Eigenarrays

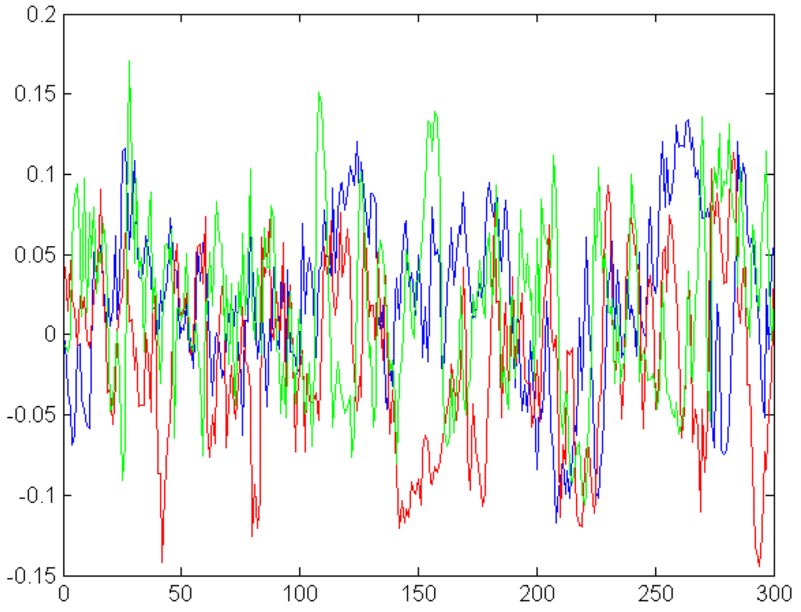
$$\mathbf{X} = \sum_{i=1}^{M} s_i \, \mathbf{U}_{\cdot i} \otimes \mathbf{V}_{\cdot i}$$

Thus far, we haven't actually "looked at" the largest-SV orthogonal basis vectors, namely the first few columns of $\mathbf{U}$ and $\mathbf{V}$

```
plot(U(:,1),'b')
hold on
plot(U(:,2),'r')
plot(U(:,3),'g')
hold off
```

```
plot(V(:,1),'b')
hold on
plot(V(:,2),'r')
plot(V(:,3),'g')
hold off
```





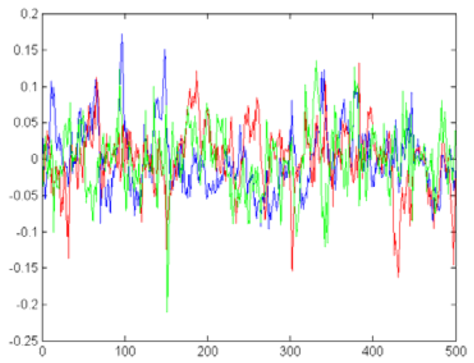
These are "eigengenes", the linear combination of genes that explain the most data.

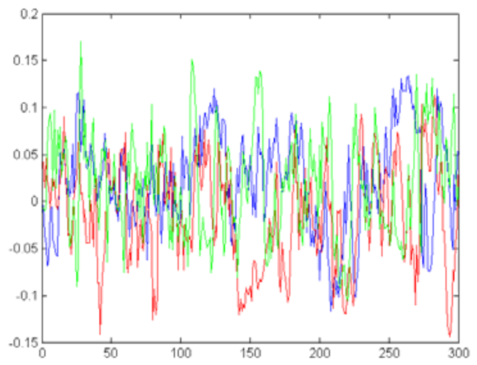
times s_i $\longleftrightarrow$

These are "eigenarrays", the linear combination of experiments that explain the most data.

However, except in special cases, eigengenes and eigenarrays are not easily interpreted.

Since we can permute the order of experiments and/or genes in the data, the “shape” of the eigenfunctions has no particular meaning here.





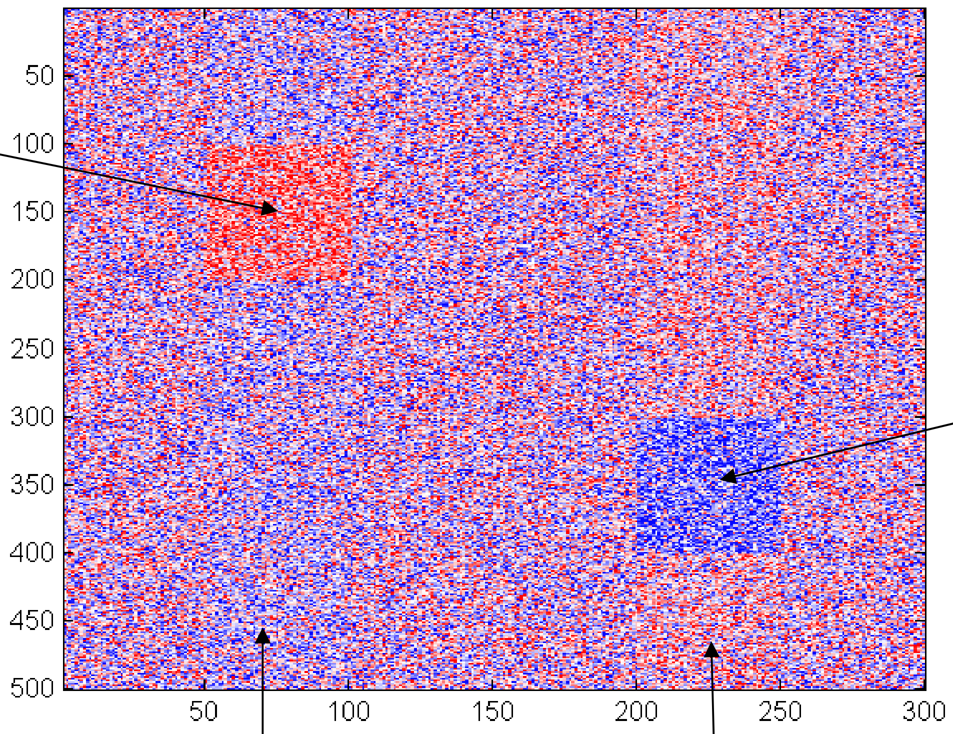
Also, as discussed before, main effects generally don’t correspond 1-to-1 to eigenanythings. At best, ~K main effects are in ~K eigenthingies.

Let’s construct a toy gene expression example with 2 main effects, and see how they show up in eigengenes and eigenarrays.

Make a toy example with (what we would call) 2 main effects:

```
pdata = randn(500,300);
pdata(101:200,51:100) = pdata(101:200,51:100) + 1;
pdata(301:400,201:250) = pdata(301:400,201:250) - 1;
pmean = mean(pdata,1);
pstd = std(pdata,1);
pdata = (pdata - repmat(pmean,[size(pdata,1),1]))./repmat(pstd,[size(pdata,1),1]);
colormap(genecolormap)
image(20*pdata+32)
```

these genes overexpress in these conditions

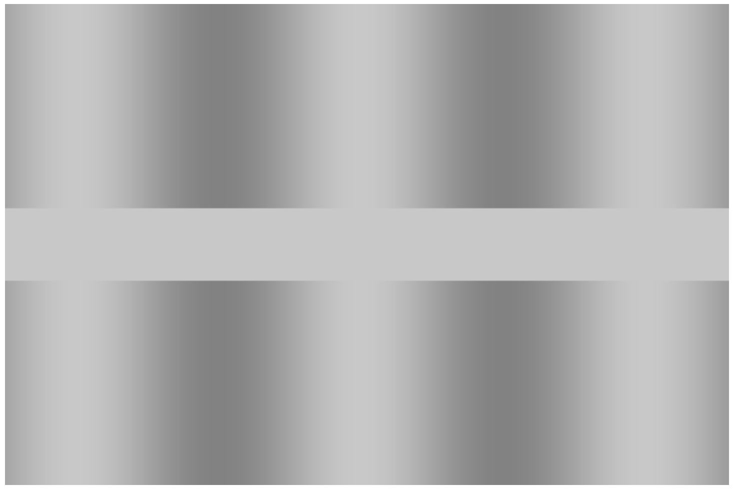


these genes underexpress in these conditions

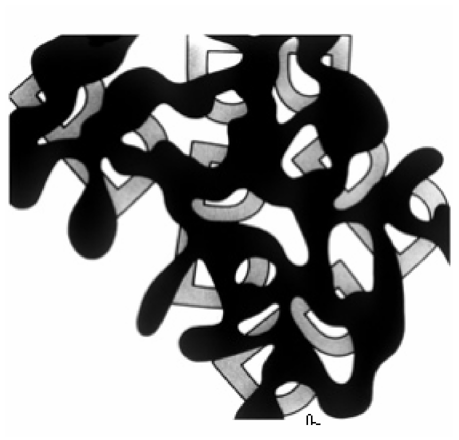
note slight vertical banding from subtracting the means

Did you see the “visual completion” or “visual phantom” illusions in the previous slide?

How about these?



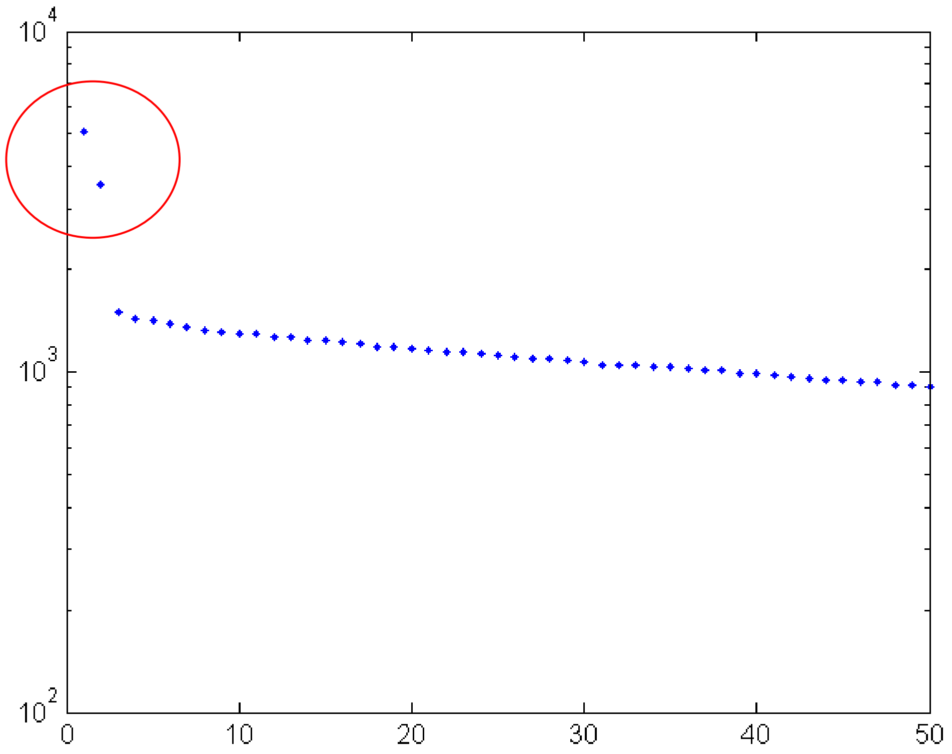
Akiyoshi Kitaoka (Ritsumeikan University)



Bregman AS (1981)

As (naively?) expected, there are exactly two large principal components (or SVs)

```
[Up Sp Vp] = svd(pdata,0);
spsq = diag(Sp).^2;
semilogy(spsq(1:50),'.b')
```



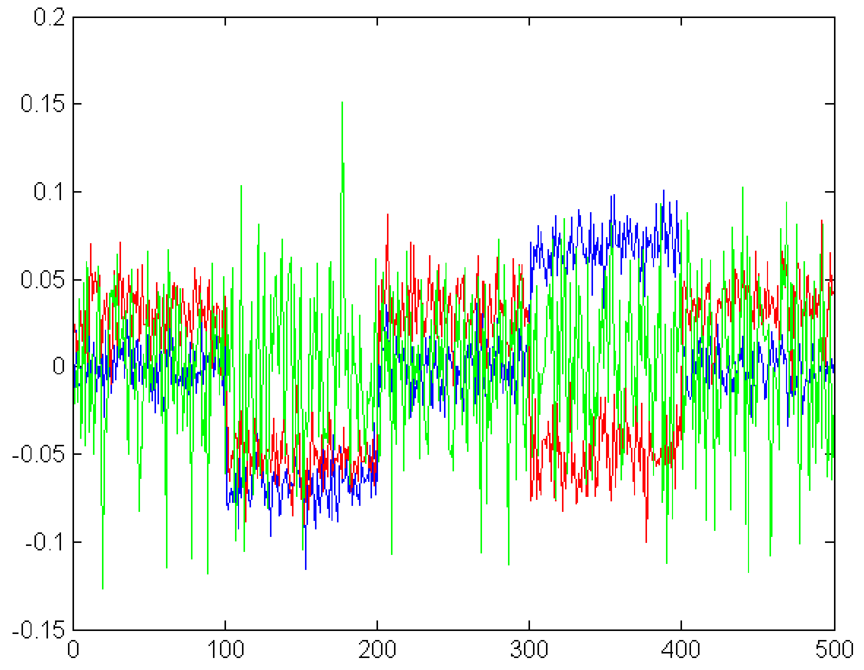
continues on to 300

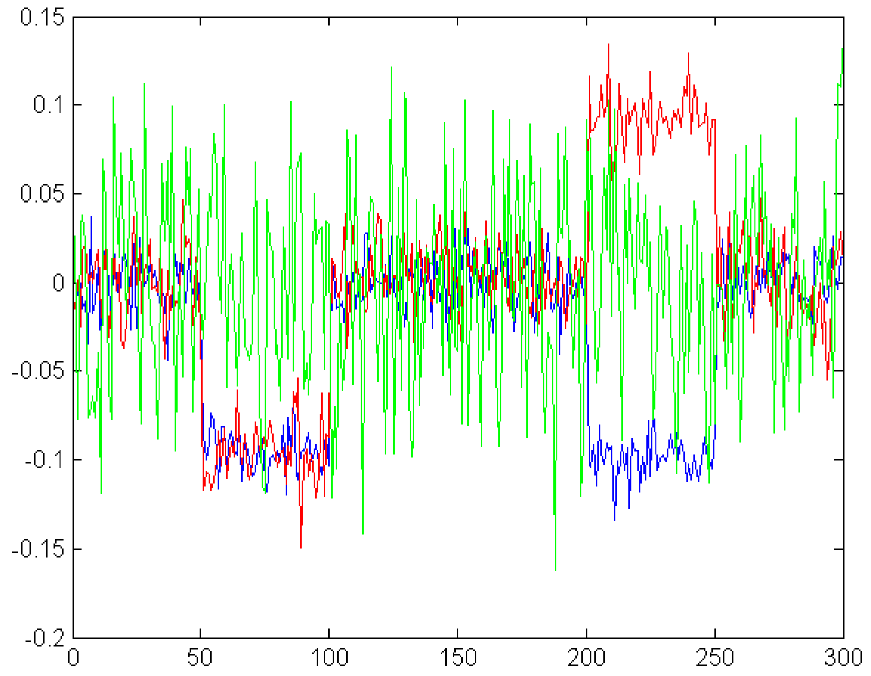
So should we expect the eigengenes/eigenarrays to show the separate main effects?

Plot 1st three eigengenes and eigenarrays

```
plot(Up(:,1),'b')
hold on
plot(Up(:,2),'r')
plot(Up(:,3),'g')
```

```
plot(Vp(:,1),'b')
hold on
plot(Vp(:,2),'r')
plot(Vp(:,3),'g')
```





First two contain an (orthogonalized) mixture of the two main effects.
Third one is, as we expect, "random".

If we had 20 main effects, the first 20 eigengenes/arrays would be mixtures of them.

Mention in passing:

There exist methods of Non-negative Matrix Factorization (NMF) whose purpose is to stop main effects from mixing when positivity matters.

Example: cluster text documents by word counts

$$\begin{array}{cc} & \text{words} \\ \text{documents} & \begin{pmatrix} \\ \text{counts} \\ \\ \end{pmatrix} \end{array} = \mathbf{X}$$

vector of 0 or pos weights of documents in *i* th cluster

vector of 0 or pos weights of words in *i* th cluster

$$\mathbf{X} \approx \sum_{i=1}^{M} \mathbf{F}_{\cdot i} \otimes \mathbf{G}_{\cdot i}$$

i.e. $\mathbf{X} \approx \mathbf{F}\mathbf{G}^T$ positive matrixes

For our problem, since genes can also be under-expressed, we would need

diagonal matrix of ±1

$$\mathbf{X} \approx \mathbf{F}\mathbf{S}\mathbf{G}^T$$

The problem with these methods is
1. The factorizations are (far from) unique!
2. The computational algorithms are little better than brute-force minimization of

$$||\mathbf{X} - \mathbf{F}\mathbf{G}^T||$$

How to find the global minimum??

3. There are other good clustering algorithms (GMMs, hierarchical, etc.)

A (binary) classifier classifies data points as + or −

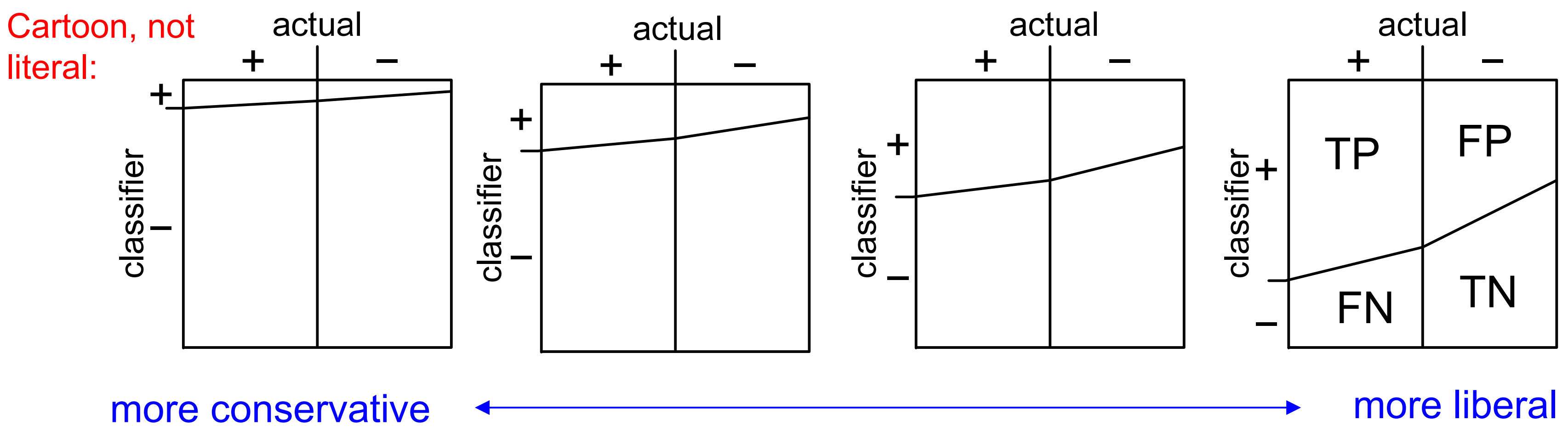
If we also know the true classification, the performance of the classifier is a 2x2 contingency table, in this application usually called a confusion matrix.

		actual +	−	
	classifier +	TP good!	FP	bad! (Type I error)
bad! (Type II error)	−	FN	TN good!	

As we saw, this kind of table has many other uses: treatment vs. outcome, clinical test vs. diagnosis, etc.

Earlier we were looking at statistically “weak” contingency tables and trying to decide if they were significant. Here we’re interested in the strength of the signal; the (high) significance is a given.

Most classifiers have a "knob" or threshold that you can adjust: How certain do they have to be before they classify a "+"? To get more TP's, you have to let in some FP's!



Notice there is just one free parameter, think of it as TP, since

FP(TP) = *[given by algorithm]*
TP + FN = P (fixed number of actual positives, column marginal)
FP + TN = N (fixed number of actual negatives, column marginal)

So all scalar measures of performance are functions of one free parameter (i.e., curves).

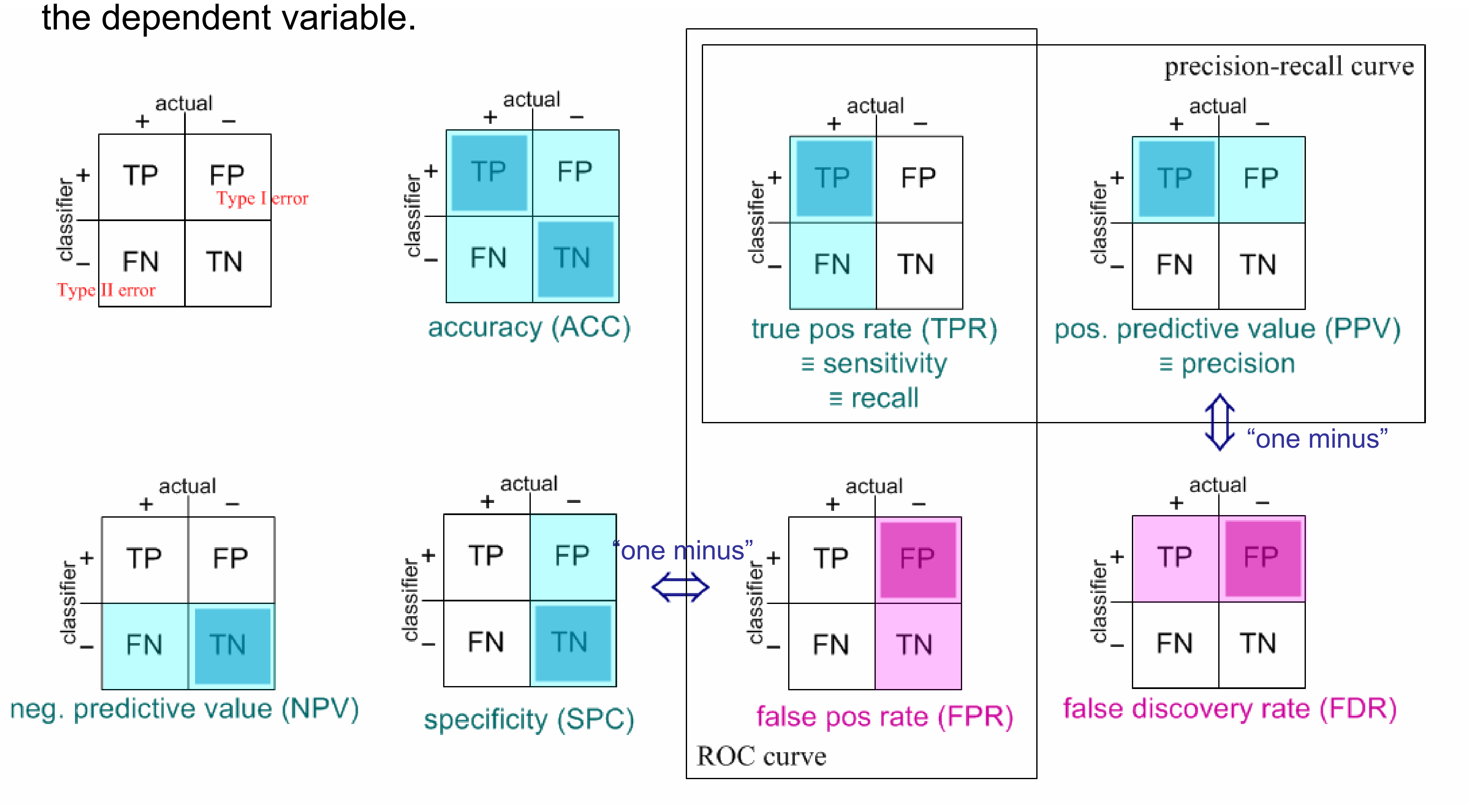
And the points on any such curve are in 1-to-1 correspondence with those on any other such curve.

If you ranked some classifiers by how good they are, you might get a different rankings at different points on the scale.

On the other hand, one classifier might dominate another at all points on the scale.

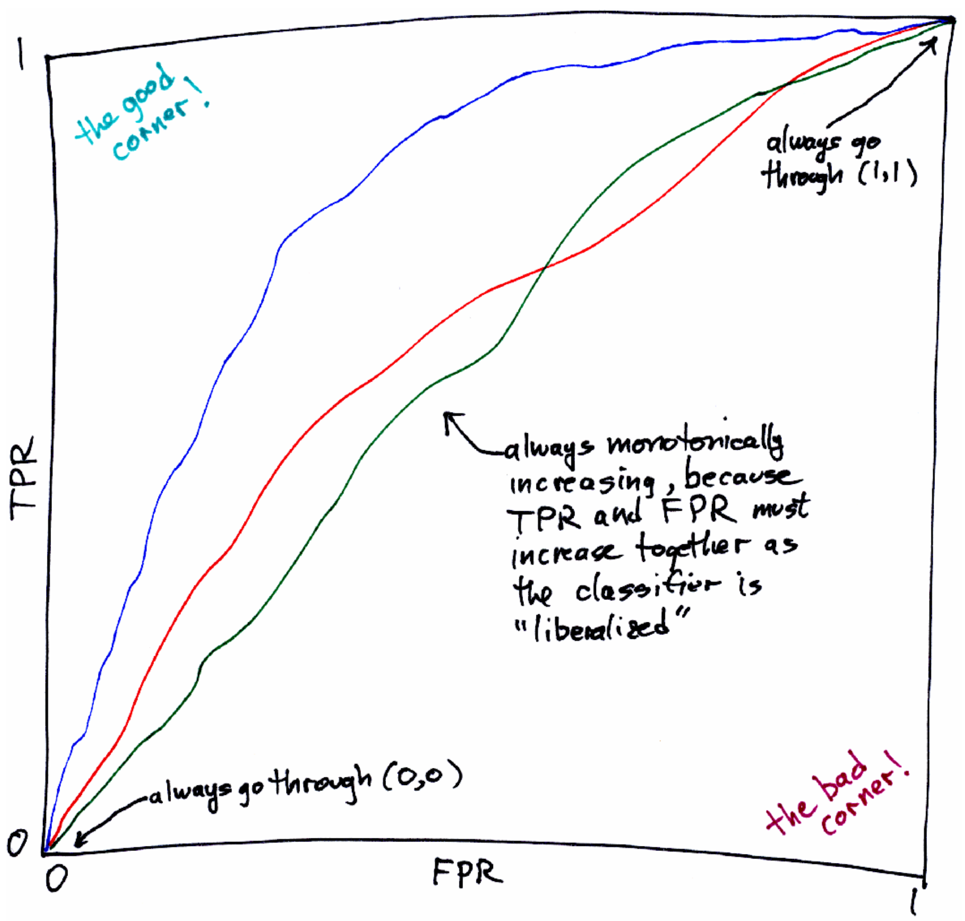
Terminology used to measure the performance of classifiers

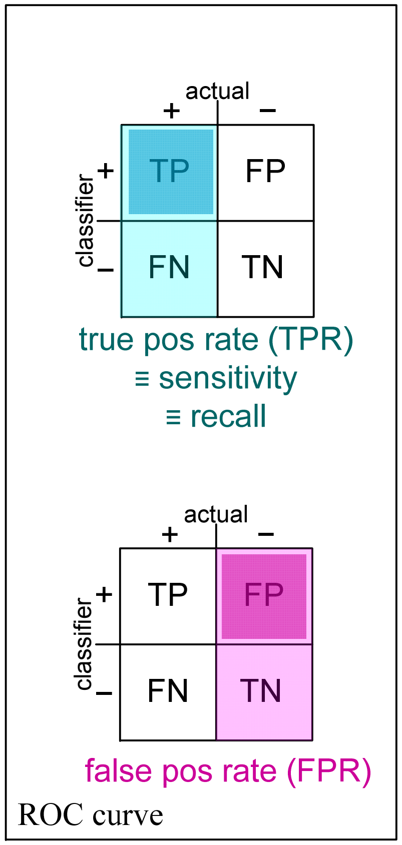
Different combinations of ratios have been given various names.
All vary between 0 and 1.
A performance curve picks one as the independent variable and looks at another as the dependent variable.



Dark color is numerator, dark and light color is denominator.
Blue parameters: 1 is good. Red: 0 is good.

ROC ("Receiver Operating Characteristic") curves plot TPR vs. FPR as the classifier goes from "conservative" to "liberal"

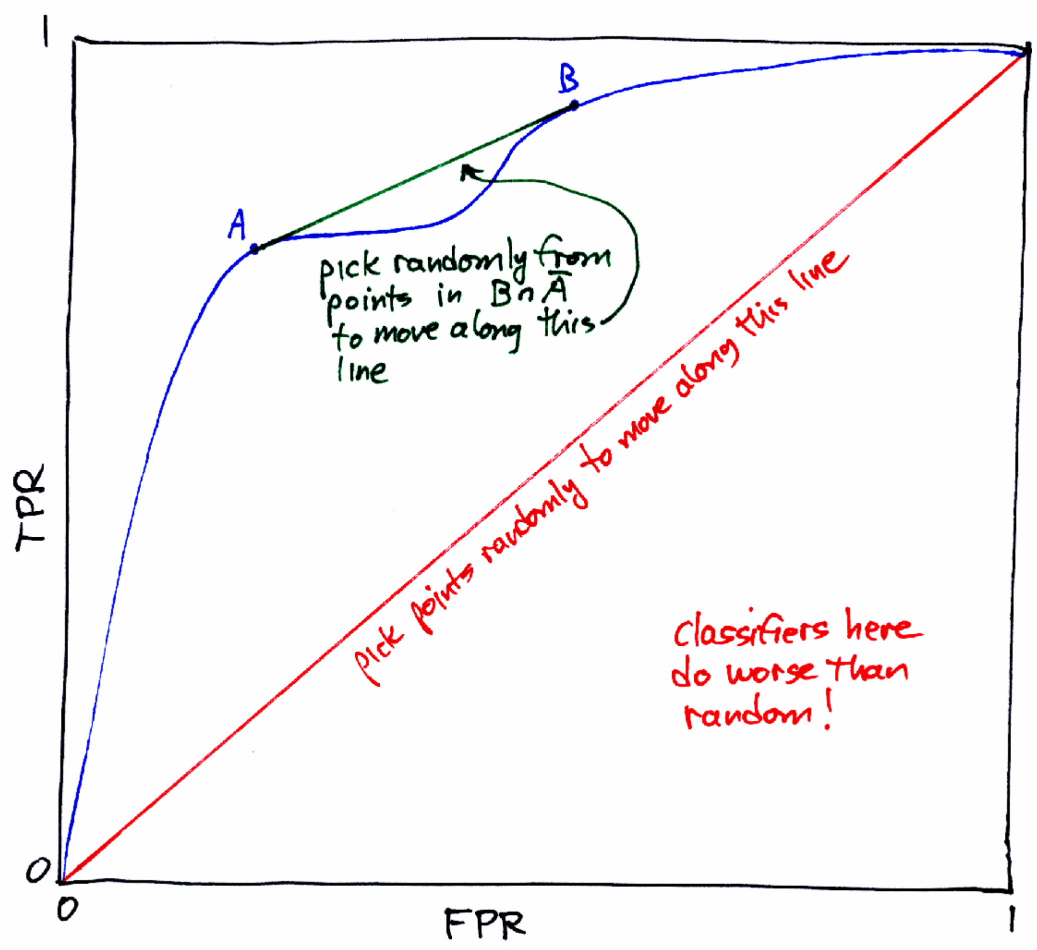




blue dominates red and green
neither red nor green dominate the other

You could get the best of the red and green curves by making a hybrid or "Frankenstein" classifier that switches between strategies at the cross-over points.

ROC curves can always be "upgraded" to their convex hull by replacing any concave portions by a random sample



List points classified as + by B but not A.

Start up the curve to A.

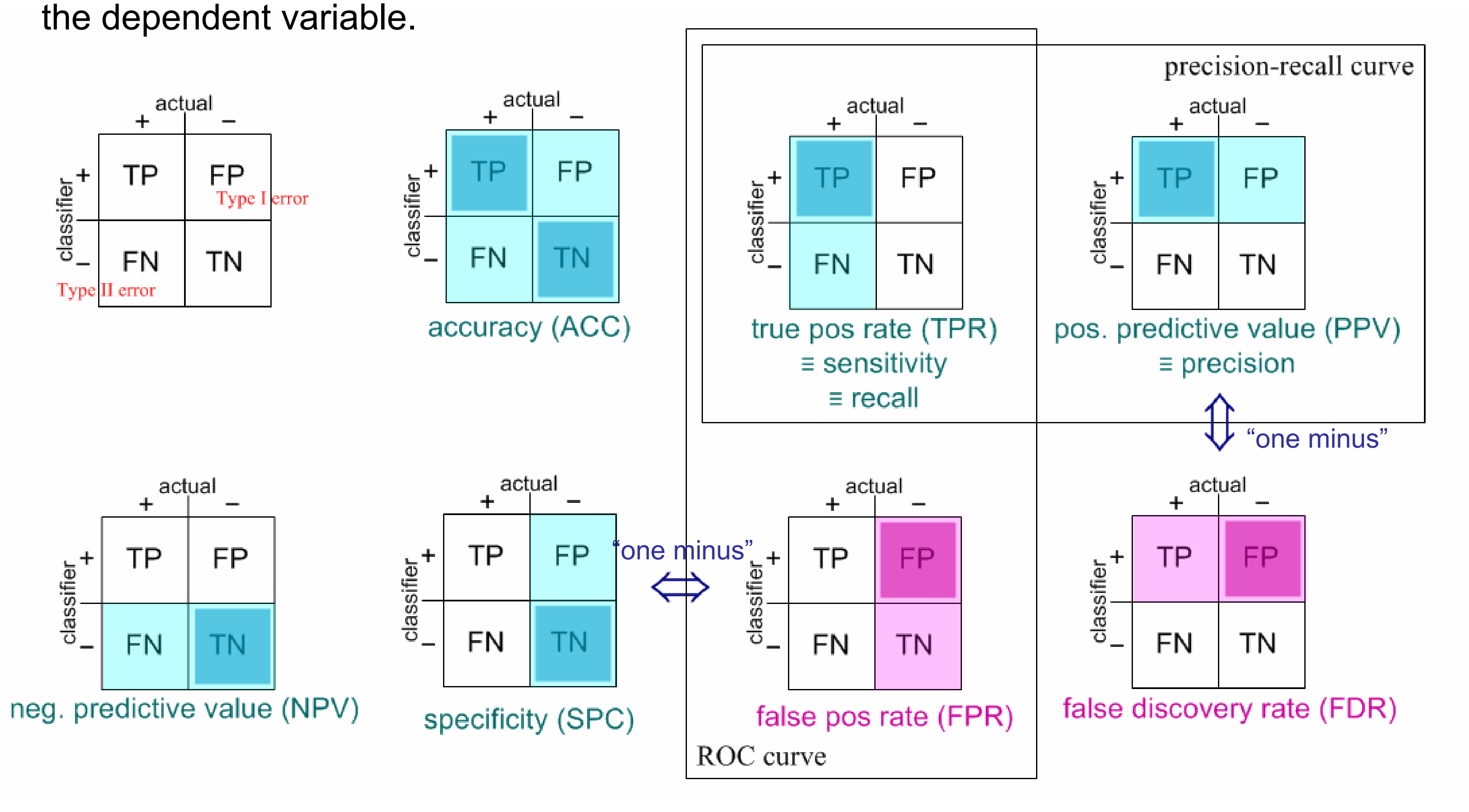
When you reach A, start adding a fraction of them (increasing from 0 to 1) randomly, until you reach B.

Continue on the curve from B.

Of course to measure the ROC curve at all, you have to have known "training" or "ground truth" data. You use that data to estimate the points A and B, then create the convex (upgraded) classifier.

Terminology used to measure the performance of classifiers

Different combinations of ratios have been given various names.
All vary between 0 and 1.
A performance curve picks one as the independent variable and looks at another as the dependent variable.

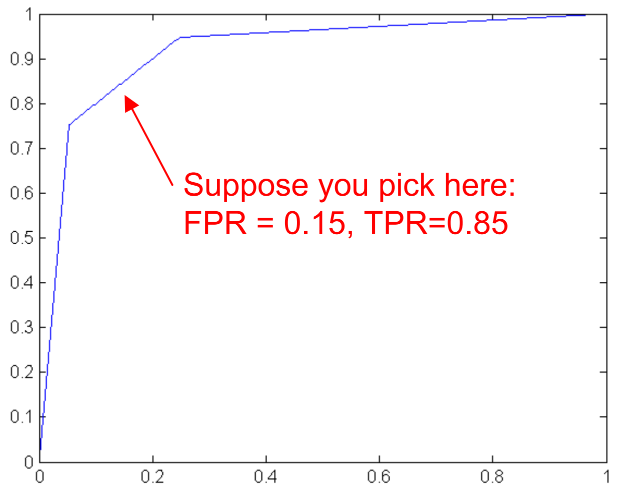


Dark color is numerator, dark and light color is denominator.
Blue parameters: 1 is good. Red: 0 is good.

Since ROC curves don't explicitly show any dependence on the constant P/N (ratio of actual + to – in the sample) they can be misleading if you care about, say, FP versus TP (or any other cross-column comparison).

Suppose you have a test for Alzheimer's whose false positive rate can be varied from 5% to 25% as the false negative rate varies from 25% to 5% (suppose linear dependences on both):

```
lam = (0:0.01:1);
fpr = .05 + 0.2 * lam;
tpr = 1 - (.05 + 0.2*(1-lam));
fpr(1) = 0;
fpr(end) = 1;
tpr(1) = 0;
tpr(end) = 1;
plot(fpr,tpr)
```



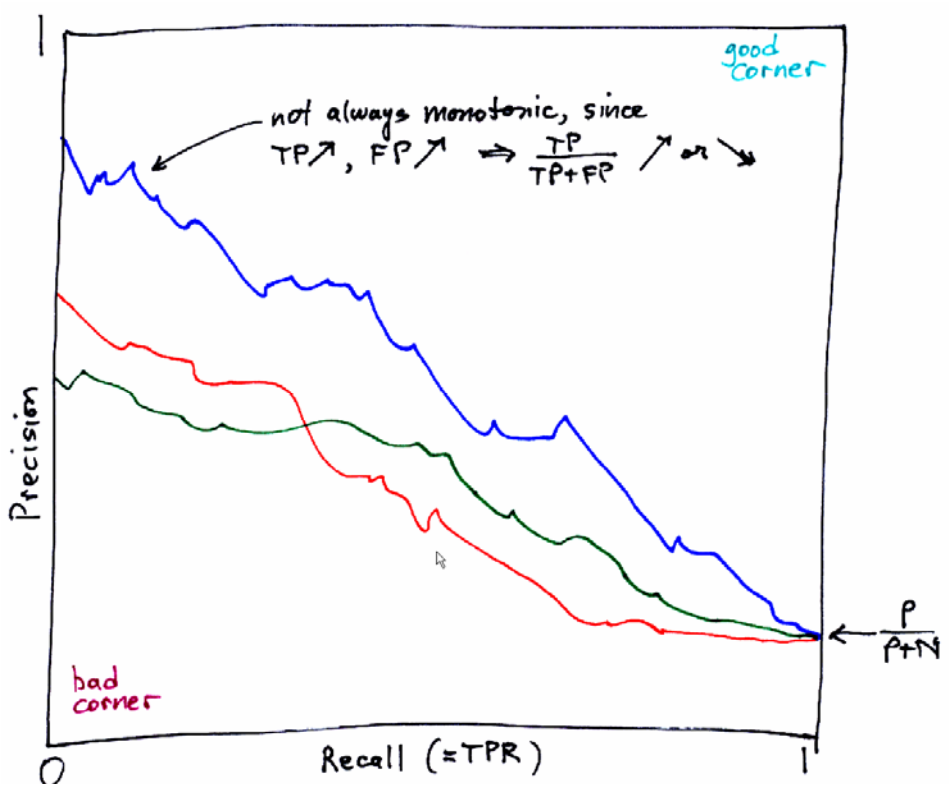
Now suppose you try the test on a population of 10,000 people, 1% of whom actually are Alzheimer's positive:

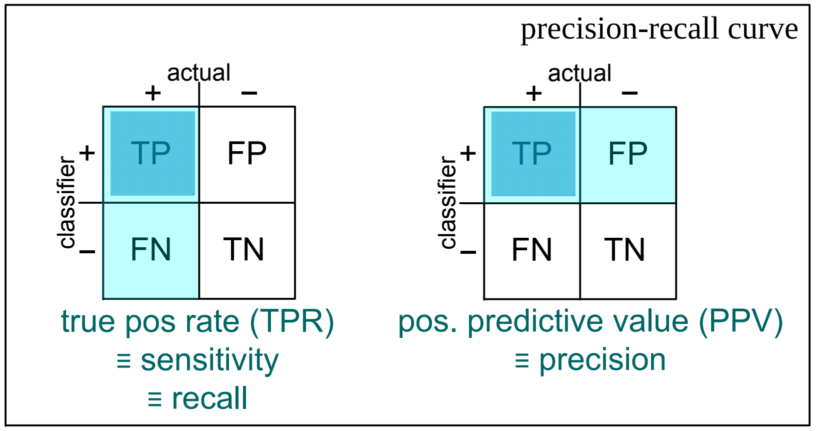
	actual +	actual –
classifier +	85	1485
classifier –	15	8415

FP swamps TP by ~17:1. You'll be telling 17 people that they might have Alzheimer's for every one who actually does. It is unlikely that your test will be used.

In a case like this, ROC, while correct, somewhat misses the point.

Precision-Recall curves overcome this issue by comparing TP with FN and FP



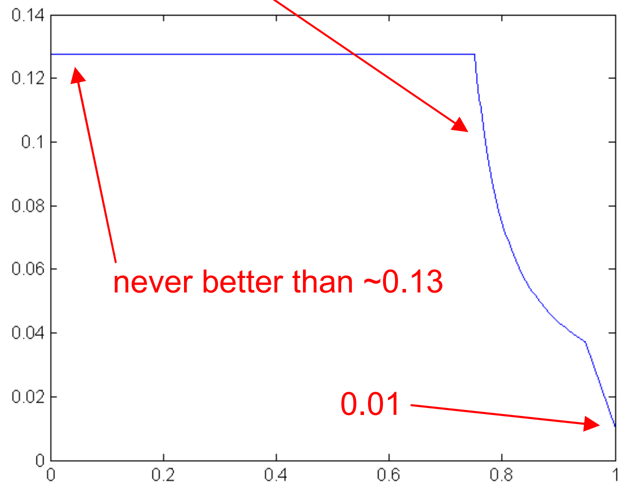


By the way, this shape "cliff" is what the ROC convexity constraint looks like in a Precision-Recall plot. It's not very intuitive.

Continue our toy example:

note that P and N now enter

```
prec = tpr*100./(tpr*100+fpr*9900);
prec(1) = prec(2); % fix up 0/0
reca = tpr;
plot(reca,prec)
```



never better than ~0.13

0.01

For fixed marginals P,N the points on the ROC curve are in 1-to-1 correspondence with the points on the Precision-Recall curve.

That is, both display the same information. You can go back and forth.

$$\left.\begin{aligned} \text{pre} &= \frac{\text{TPR}\,P}{\text{TPR}\,P + \text{FPR}\,N} \\ \text{rec} &= \text{TPR} \end{aligned}\right\}$$

pre, rec from TPR, FPR

$$\left\{\frac{\text{rec}\,(1-\text{pre})}{\text{pre}}\,\frac{P}{N} = \text{FPR}\right.$$

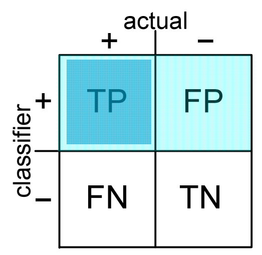
TPR, FPR from pre, rec

It immediately follows that if one curve dominates another in ROC space, it also dominates in Precision-Recall space.

(Because a crossing in one implies a crossing in the other, by the above equations.)

But for curves that cross, the metrics in one space don't easily map to the other. For example, people sometimes use "area under the ROC curve". This doesn't correspond to "area under the Precision-Recall curve", or to anything simple.

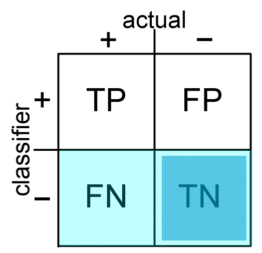
One also sees used PPV and NPV
(more often as a pair of numbers than as a curve)



pos. predictive value (PPV)
≡ precision

PPV: given a positive test, how often does the patient have the disease.

NPV: given a negative test, how often is the patient disease-free.



neg. predictive value (NPV)

classifier \ actual	+	−
+	85	1485
−	15	8415

PPV = 0.054

NPV = 0.998

So could a physician use this test “to rule out Alzheimers” in a case that presents with some symptoms?

No, because in the population of people who present, the ratio of the columns would be not nearly so extreme.

You have to be careful about asking exactly the question you want!

It's easy to get from PPV,NPV to ROC or vice versa. Or, for that matter, any other of the parameterizations. In Mathematica, for example:

```
In[1]:= eqs = {PPV == TP / (TP + FP), NPV == TN / (TN + FN), TP + FN == P, TN + FP == N}
```

Out[1]= $\left\{\mathrm{PPV}==\frac{\mathrm{TP}}{\mathrm{FP}+\mathrm{TP}},\ \mathrm{NPV}==\frac{\mathrm{TN}}{\mathrm{FN}+\mathrm{TN}},\ \mathrm{FN}+\mathrm{TP}==\mathrm{P},\ \mathrm{FP}+\mathrm{TN}==\mathrm{N}\right\}$

```
In[2]:= ans = FullSimplify[Solve[eqs, {TP, FP, TN, FN}]]
```

Out[2]= $$\left\{\left\{\mathrm{FN}\to-\frac{(-1+\mathrm{NPV})\ (\mathrm{P}\ (-1+\mathrm{PPV})+\mathrm{N}\ \mathrm{PPV})}{-1+\mathrm{NPV}+\mathrm{PPV}},\ \mathrm{TP}\to\frac{(\mathrm{N}\ (-1+\mathrm{NPV})+\mathrm{NPV}\ \mathrm{P})\ \mathrm{PPV}}{-1+\mathrm{NPV}+\mathrm{PPV}},\right.\right.$$
$$\left.\left.\mathrm{FP}\to-\frac{(\mathrm{N}\ (-1+\mathrm{NPV})+\mathrm{NPV}\ \mathrm{P})\ (-1+\mathrm{PPV})}{-1+\mathrm{NPV}+\mathrm{PPV}},\ \mathrm{TN}\to\frac{\mathrm{NPV}\ (\mathrm{P}\ (-1+\mathrm{PPV})+\mathrm{N}\ \mathrm{PPV})}{-1+\mathrm{NPV}+\mathrm{PPV}}\right\}\right\}$$

```
In[4]:= TPR = FullSimplify[TP / (TP + FN) /. ans]
```

Out[4]= $\left\{\frac{(\mathrm{N}\ (-1+\mathrm{NPV})+\mathrm{NPV}\ \mathrm{P})\ \mathrm{PPV}}{\mathrm{P}\ (-1+\mathrm{NPV}+\mathrm{PPV})}\right\}$

```
In[5]:= FPR = FullSimplify[FP / (FP + TN) /. ans]
```

Out[5]= $\left\{-\frac{(\mathrm{N}\ (-1+\mathrm{NPV})+\mathrm{NPV}\ \mathrm{P})\ (-1+\mathrm{PPV})}{\mathrm{N}\ (-1+\mathrm{NPV}+\mathrm{PPV})}\right\}$

In[8]:= $\mathbf{eqs2} = \left\{\mathtt{tpr} == \left(\frac{(\mathrm{N}\ (-1+\mathrm{NPV})+\mathrm{NPV}\ \mathrm{P})\ \mathrm{PPV}}{\mathrm{P}\ (-1+\mathrm{NPV}+\mathrm{PPV})}\right),\ \mathtt{fpr} == \left(-\frac{(\mathrm{N}\ (-1+\mathrm{NPV})+\mathrm{NPV}\ \mathrm{P})\ (-1+\mathrm{PPV})}{\mathrm{N}\ (-1+\mathrm{NPV}+\mathrm{PPV})}\right)\right\}$

Out[8]= $\left\{\mathtt{tpr}==\frac{(\mathrm{N}\ (-1+\mathrm{NPV})+\mathrm{NPV}\ \mathrm{P})\ \mathrm{PPV}}{\mathrm{P}\ (-1+\mathrm{NPV}+\mathrm{PPV})},\ \mathtt{fpr}==-\frac{(\mathrm{N}\ (-1+\mathrm{NPV})+\mathrm{NPV}\ \mathrm{P})\ (-1+\mathrm{PPV})}{\mathrm{N}\ (-1+\mathrm{NPV}+\mathrm{PPV})}\right\}$

```
In[9]:= FullSimplify[Solve[eqs2, {NPV, PPV}]]
```

Out[9]= $\left\{\left\{\mathrm{NPV}\to\frac{(-1+\mathtt{fpr})\ \mathrm{N}}{(-1+\mathtt{fpr})\ \mathrm{N}+\mathrm{P}\ (-1+\mathtt{tpr})},\ \mathrm{PPV}\to\frac{\mathrm{P}\ \mathtt{tpr}}{\mathtt{fpr}\ \mathrm{N}+\mathrm{P}\ \mathtt{tpr}}\right\}\right\}$