Statistical methods for high-dimensional biology

STAT 540 2016

Data exploration

Paul Pavlidis

Project pitches

Our prototype data set

Each value =
Measurement of one gene one sample

One of these gives us measurements for every gene, in a sample

One column per sample

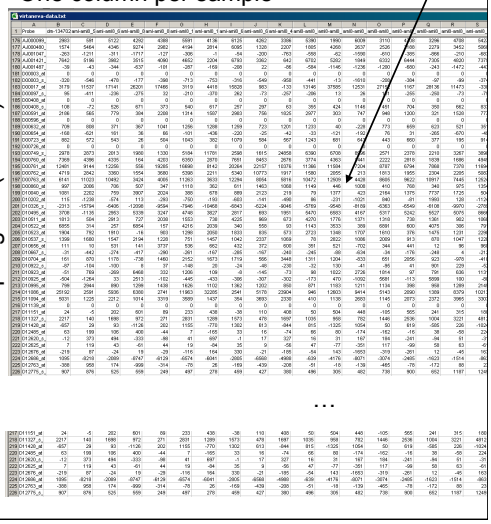
Samples

Case type A

Assays

Case type B

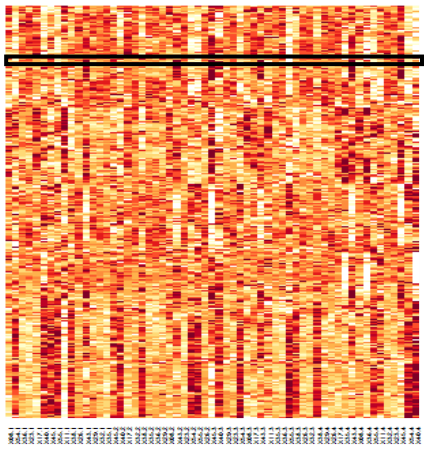
One row per gene (20000)

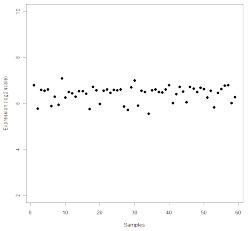


Profile for a gene

Samples

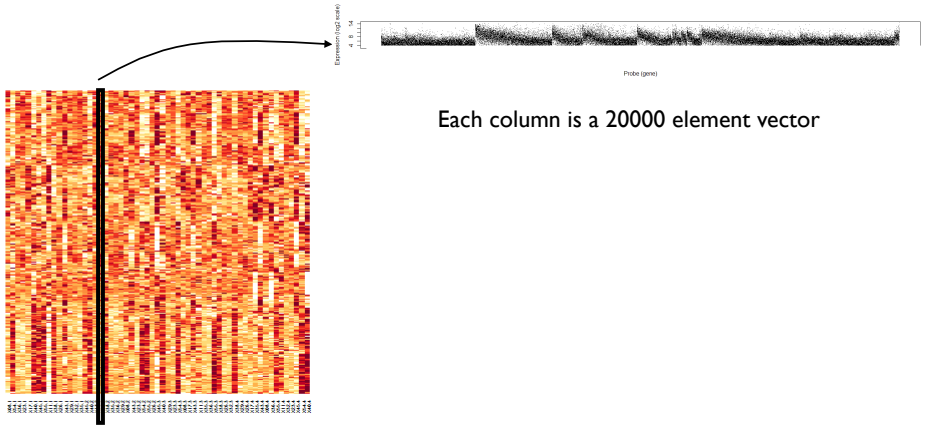
Gene





Profile for a gene. This is a 59-element vector

Profile for a sample



Each column is a 20000 element vector

(for many, many more rows)

*This is a schematic. The graph and color map don't match

What you usually get

Data file: contains the measured data

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	probe	8.1	54.1	36.1	23.1	17.1	40.1	45.1	55.1	11.1	38.1	26.1	43.1	29.1	32.1	35.1	45.
2	1007_s_at	7.926	7.0559	7.9411	7.7977	7.4689	7.5711	7.4303	7.2057	7.4715	7.1806	8.2236	7.2317	8.0646	7.73	8.0793	7.662
3	1053_at	6.6365	7.1358	7.5862	7.8645	7.4806	7.3865	7.9948	7.4433	6.5344	7.503	7.7826	7.0135	7.9657	8.3421	7.8595	7.7
4	117_at	8.959	8.4803	10.105	10.271	7.9707	10.321	10.456	10.382	8.3655	9.6068	9.7467	8.7966	9.4911	10.661	9.8787	10.33
5	121_at	8.6827	8.3495	8.9023	9.2696	8.8656	8.2199	8.5431	8.4339	9.2383	8.3834	9.0861	8.7765	8.9787	8.9857	9.2812	8.669
6	1255_g_at	3.9677	3.8677	3.9605	3.9194	3.9049	3.8048	3.7421	3.6676	3.9594	3.9285	3.9448	3.8905	4.024	4.0583	4.0189	3.960
7	1294_at	8.6788	9.0819	9.0982	8.8084	8.0572	9.1845	9.0646	9.3731	8.0459	9.3914	9.1369	8.7087	9.2214	9.4563	8.9616	9.332
8	1316_at	5.9764	5.8148	5.8111	5.9711	5.8282	5.9679	5.6319	5.6115	5.9962	5.9118	5.9765	5.9705	6.0618	5.8713	5.8805	5.73
9	1320_at	5.167	5.1309	5.2289	5.1778	5.2646	4.6999	4.9622	5.0373	5.3104	4.7154	5.1481	5.0763	5.2984	5.2667	5.1752	4.987
10	1405_i_at	6.2639	7.5861	6.7248	5.8717	6.1697	10.464	6.0142	5.6844	5.89	5.677	6.0243	6.1046	7.5059	7.5744	8.795	5.608
11	1431_at	4.1273	3.8827	4.2965	4.1919	4.0972	4.1824	3.899	3.871	4.2915	3.8586	4.1631	3.9662	4.0597	4.1489	4.2477	3.929
12	1438_at	6.7376	5.9663	6.7058	6.8065	6.8481	6.1072	6.1491	5.8832	6.9704	6.4975	6.6712	6.6884	6.6617	6.6978	6.6162	6.736
13	1487_at	8.1541	7.9911	8.449	8.6151	7.9772	7.9992	8.5637	7.6781	7.9142	8.5596	8.6313	8.0206	8.5319	8.489	8.5775	8.713
14	1494_f_at	6.3542	5.6148	6.5204	6.5367	6.3581	5.695	6.314	6.1736	6.8538	5.7591	6.241	6.3872	6.4009	6.3022	6.7324	6.033
15	1552256_a	8.2499	7.337	7.9888	8.1884	7.624	6.9916	7.7671	7.673	7.5446	8.2511	7.9856	7.6377	8.1844	8.3138	8.2106	7.464
16	1552257_a	7.8619	8.6819	7.7546	7.7172	7.1525	7.2768	7.2925	8.6039	7.6841	7.7207	7.4003	7.17	7.9763	7.6881	7.6714	7.314
17	1552258_a	6.3882	6.2678	5.9786	6.033	5.9467	5.8941	5.8066	5.7953	6.4704	5.5054	5.8349	5.9561	6.0665	5.8455	5.9079	5.604
18	1552260_s	5.2138	5.4845	5.2098	5.2086	5.3298	4.8697	4.758	5.1423	5.3708	5.4289	5.1494	5.9599	4.9137	5.1657	5.035	5.254
19	1552263_s	8.1053	8.8008	8.5589	8.5378	8.2161	8.6552	9.1113	9.1421	7.3856	9.1848	9.0102	7.9257	9.142	8.9135	8.6977	8.971
20	1552264_a	8.6917	8.1056	9.1256	9.5827	8.5806	8.9591	9.6586	9.1135	7.8937	10.159	9.5357	8.8835	9.6652	9.597	9.068	9.804
21	1552266_s	4.6685	4.4884	4.7066	4.824	4.8375	4.4521	4.5001	4.3423	5.2833	4.4172	4.7029	4.686	4.5755	4.6582	4.6378	4.782
22	1552269_s	4.4192	4.2144	4.437	4.5913	4.3935	4.4655	4.4572	4.4083	4.9316	4.4265	4.3362	4.7248	4.437	4.3781	4.4576	4.503
23	1552270_s	6.3119	6.345	6.3908	6.2609	6.2344	6.5756	6.2433	6.3708	6.2118	6.1392	6.2828	7.1272	6.4115	6.3586	6.6803	6.56
24	1552272_s	5.394	5.5444	5.7037	6.0072	5.5832	5.3515	5.4648	5.0233	5.6239	5.6021	5.6298	5.4619	5.6787	5.6563	5.7104	5.361
25	1552274_s	8.3238	8.3465	7.7187	7.5153	7.7277	7.4438	7.2911	8.2899	7.4731	7.9592	7.7983	7.178	7.8233	7.8738	7.5345	7.195
26	1552275_s	8.2882	8.3604	8.4839	8.4818	8.8081	8.1993	8.179	8.3606	7.1891	9.2228	8.5196	7.7405	8.8754	8.8907	8.2822	8.112
27	1552276_a	6.7279	6.0901	6.695	6.683	6.2504	6.6675	6.5971	6.4836	6.2384	6.5713	6.4075	6.8016	6.749	6.6906	6.7675	6.653
28	1552277_s	5.5806	5.4087	5.6828	6.1552	5.8643	5.5927	5.7728	5.5492	6.0694	5.5681	5.7958	5.2635	6.068	5.9254	6.0306	5.428
29	1552278_a	5.1844	4.93	5.0426	5.151	5.2738	4.8416	4.8104	4.8391	5.3889	5.0153	5.0466	5.0051	4.8121	4.7676	5.1427	4.70
30	1552279_s	7.3871	6.9362	7.239	7.5541	7.237	6.7551	6.8717	7.0931	7.2939	7.1352	7.2133	6.9912	7.0548	7.2409	7.41	6.979
31	1552280_a	4.9163	4.653	5.1365	5.1372	5.3321	4.683	4.5319	4.4803	5.5901	4.8264	4.7986	5.7601	4.8146	4.8212	5.1236	4.646
32	1552281_a	8.1083	7.6469	8.0105	8.1162	8.1639	7.9042	7.7853	7.4942	8.259	7.7484	7.9145	8.0105	8.0369	8.0843	8.3915	8.127
33	1552283_s	4.5335	4.546	4.6218	4.3103	4.5545	4.8776	4.4279	4.1352	4.6874	4.6024	4.4879	4.4859	4.3427	4.4104	4.4026	4.759
34	1552286_a	7.7831	8.1449	7.0758	7.4632	7.7199	6.7557	6.8364	7.7087	7.4367	7.3802	7.3676	6.6369	7.1095	7.3945	7.4114	7.184
35	1552287_a	8.6704	9.5144	7.7426	7.3035	7.6311	8.2642	7.6958	9.8955	8.6347	8.0455	7.6622	6.9434	7.9512	7.9819	7.6845	7.26
36	1552288_s	4.7877	4.5473	4.6671	4.4615	4.6667	4.5701	4.6778	4.5328	4.7977	4.666	4.3686	4.6386	4.5223	4.5467	4.5473	4.445
37	1552289_a	5.5383	4.7204	5.2587	5.2168	5.4514	4.6964	4.9417	4.7889	5.7215	4.9387	5.3342	5.0251	4.8955	5.0112	5.2695	5.206
38	1552291_s	7.1732	7.1967	6.6759	6.7014	8.2649	7.4813	6.8194	7.9149	7.1002	7.2438	7.1017	6.3061	6.9156	7.0574	6.7851	5.948
39	1552293_s	5.4462	5.1998	5.5808	5.1741	5.4695	5.1185	5.2217	4.8742	5.4265	5.387	5.6759	5.126	5.432	5.4098	5.6203	5.358
40	1552295_a	6.7185	6.4899	7.066	7.2199	7.3332	6.9241	6.8265	7.0179	7.029	7.3956	7.0454	7.1157	7.2581	7.1137	7.2951	6.992

Metadata file: describes the samples

Pat	Patient	Sample	Affected	Diagnosis
1.1	1	1	Yes	1: Chronic
1.2	1	2	Yes	1: Chronic
1.4	1	4	No	1: Chronic
2.1	2	1	Yes	2: Aggressive
2.2	2	2	Yes	2: Aggressive
2.4	2	4	No	2: Aggressive
3.1	3	1	Yes	2: Aggressive
3.2	3	2	Yes	2: Aggressive
3.3	3	3	No	2: Aggressive
4.1	4	1	Yes	1: Chronic
4.2	4	2	Yes	1: Chronic
4.3	4	3	No	1: Chronic
5.1	5	1	No	1: Chronic
5.2	5	2	Yes	1: Chronic
5.3	5	3	Yes	1: Chronic
7.1	7	1	Yes	2: Aggressive
7.2	7	2	Yes	2: Aggressive
7.3	7	3	Yes	2: Aggressive
8.1	8	1	Yes	1: Chronic
8.2	8	2	Yes	1: Chronic
8.3	8	3	No	1: Chronic

a.k.a. design file, sample information file

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	probe	8.1	54.1	36.1	23.1	17.1	40.1	45.1	55.1	11.1	38.1	26.1	43.1	29.1	32.1	35.1	45.
2	1007_s_at	7.926	7.0559	7.9411	7.7977	7.4689	7.5711	7.4303	7.2057	7.4715	7.1906	8.2236	7.2317	8.0644	7.73	8.0763	7.642
3	1053_at	6.6365	7.1358	7.5862	7.8645	7.4806	7.3865	7.9948	7.4433	6.5344	7.503	7.7826	7.0135	7.9657	8.3421	7.8595	7.7
4	117_at	8.959	8.4803	10.105	10.271	7.9707	10.321	10.456	10.382	8.5655	9.6068	9.7467	8.7966	9.4911	10.661	9.8787	10.35
5	121_at	8.6827	8.3495	8.9823	9.2696	8.8456	8.2199	8.5431	8.4339	9.2383	8.3634	9.0861	8.7765	8.9787	8.9857	9.2812	8.669
6	1255_g_at	3.9677	3.8677	3.9605	3.9194	3.9049	3.8048	3.7421	3.6676	3.9594	3.9285	3.9448	3.8905	4.024	4.0583	4.0189	3.960
7	1294_at	8.6788	9.0819	9.0982	8.8084	8.0572	9.1845	9.0646	9.3731	8.0459	9.5914	9.1369	8.7087	9.2214	9.4563	8.9616	9.332
8	1316_at	5.9764	5.8148	5.8111	5.9711	5.8282	5.9679	5.6319	5.6115	5.9962	5.9318	5.9918	5.5705	6.0638	5.8713	5.8805	5.73
9	1320_at	5.167	5.1309	5.2289	5.1778	5.2646	4.6999	4.9622	5.0373	5.5104	4.7154	5.1481	5.0763	5.2984	5.2667	5.1752	4.987
10	1405_i_at	6.2639	7.5861	6.7248	5.81	6.697	10.464	6.0142	5.8844	6.[illegible]	5.677	6.0[illegible]	6.1046	7.5059	7.5744	8.795	5.608
11	1431_at	4.1273	3.8827	4.2963	4.15	4.[illegible]	[illegible]899	[illegible]29	[illegible]86	4.1531	3.9682	4.0597	4.1489	4.2477	3.929		
12	1438_at	6.7376	5.9663	6.7058	6.84[illegible]	[illegible]481	[illegible]107	[illegible]1491	[illegible]865	[illegible]971	[illegible]75	[illegible]12	6.6884	6.6617	6.6978	6.6162	6.730
13	1487_at	8.1541	7.9911	8.449	8.6151	7.9772	7.9992	8.5637	7.6781	7.9142	8.5596	8.6313	8.0206	8.5319	8.489	8.5775	8.712
14	1494_f_at	6.3542	5.6148	6.5304	6.5367	6.3581	5.695	6.314	6.1736	6.8538	5.7593	6.241	6.3872	6.4009	6.3022	6.7324	6.033
15	1552256_a	8.2499	7.537	7.9888	8.3884	7.624	6.9916	7.9671	7.673	7.5446	8.2511	7.9856	7.6377	8.1844	8.3138	8.2106	7.464
16	1552257_a	7.8619	8.6819	7.7546	7.7172	7.1525	7.2768	7.2925	8.6039	7.6841	7.7207	7.4003	7.17	7.9763	7.6881	7.6714	7.314
17	1552258_a	6.3882	6.2678	5.9786	6.053	5.9467	5.8941	5.8086	5.7953	6.4704	5.5054	5.8349	5.9561	6.0665	5.8455	5.9879	5.604
18	1552261_a	5.2138	5.4845	5.2098	5.2086	5.3298	4.8697	4.758	5.3423	5.3708	5.4289	5.1434	5.9599	4.9137	5.1657	5.035	5.254
19	1552263_a	8.1053	8.8008	8.5589	8.5378	8.2161	8.6552	9.1113	9.1421	7.3856	9.1848	9.0102	7.9257	9.142	8.9135	8.6977	8.971
20	1552264_a	8.6917	8.1056	9.1256	9.5827	8.5806	8.9591	9.6586	9.1135	7.8937	10.159	9.5357	8.8835	9.6652	9.597	9.068	9.804
21	1552266_a	4.6685	4.4884	4.7066	4.824	4.8375	4.4523	4.5001	4.3423	5.2833	4.6172	4.7029	4.686	4.5755	4.6582	4.6378	4.762
22	1552269_a	4.4192	4.2144	4.437	4.5913	4.3935	4.4655	4.4572	4.4083	4.9316	4.4265	4.3362	4.7248	4.437	4.3781	4.4576	4.503
23	1552271_a	6.3119	6.545	6.3908	6.2609	6.2344	6.3756	6.2433	6.3708	6.2118	6.1392	6.2828	7.1272	6.4115	6.3586	6.6803	6.58
24	1552272_a							5.4646	5.0233	5.[illegible]	Pat	Patient	Sample	Affected_	Diagnosis		5.361
25	1552274_a							7.2911	8.2899	7.[illegible]	1.1	1	1	Yes	1: Chronic		7.195
26	1552275_s							8.179	8.3606	7.[illegible]	1.2	1	2	Yes	1: Chronic		8.113
27	1552276_a							6.5971	6.4836	6.[illegible]	1.4	1	4	No	1: Chronic		6.632
28	1552277_a							5.7728	5.5492	6.[illegible]	2.1	2	1	Yes	2: Aggressive		5.428
29	1552278_a							4.8104	4.8391	5.[illegible]	2.2	2	2	Yes	2: Aggressive		4.70
30	1552279_a							6.8717	7.0931	7.[illegible]	2.4	2	4	No	2: Aggressive		6.970
31	1552280_a							4.5319	4.4803	5.[illegible]	3.1	3	1	Yes	2: Aggressive		4.646
32	1552281_a							7.7853	7.4942	8[illegible]	3.2	3	2	Yes	2: Aggressive		8.127
33	1552283_s							4.4279	4.1352	4.[illegible]	3.3	3	3	No	2: Aggressive		4.739
34	1552286_a							6.8364	7.7087	7.[illegible]	4.1	4	1	Yes	1: Chronic		7.184
35	1552287_s							7.6958	9.8955	8.[illegible]	4.2	4	2	Yes	1: Chronic		7.26
36	1552288_a							4.6778	4.5328	4.[illegible]	4.3	4	3	No	1: Chronic		4.449
37	1552289_a							4.9417	4.7889	5.[illegible]	5.1	5	1	No	1: Chronic		5.206
38	1552291_a							6.8194	7.9149	7.[illegible]	5.2	5	2	Yes	1: Chronic		5.948
39	1552293_a							5.2217	4.8742	5.[illegible]							5.358
40	1552295_a							6.8265	7.0179	7[illegible]							6.992

Now what?

Ready for exploration

- Understand/get a feel for the data
- Formulate hypotheses / develop models
- Identify problems

The biggest mistake in data analysis

If you don't look at the data … you are going to get in trouble.

- Not just at the beginning, but at every stage.
- That could mean making graphs, or staring at numbers. Probably both.
- "Sanity checks" should make up a lot of your effort.

What are some "first questions" you might ask?

Some suggestions for first questions

- Is the file the expected size? Format?
- Is the data numeric? Integers or decimal?
- Do we have the expected number of samples?
  - Are there extra columns/rows in either file?
- Do the sample names in both files match?
  - Do not assume same ordering!
- Do we have all the sample information we want?
- Do factors have the expected number of levels?
- Are the data row names in a format I can use
  - Gene names, probe identifiers, etc.
  - Watch for R mangling names
- Are there missing data points?
- Are the data on a log scale? If so what is the base?

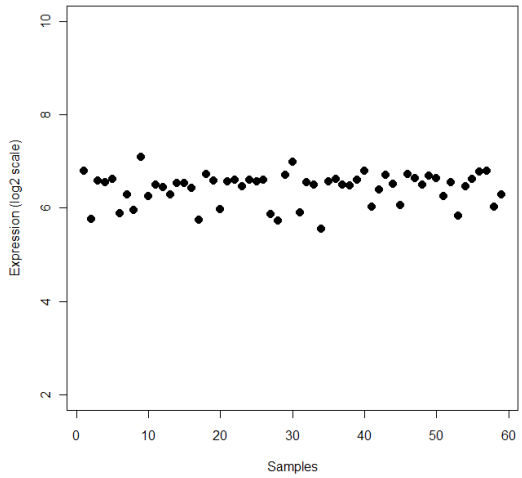
If you are the one generating the data, save yourself grief by paying attention to these issues up front.

Document what you did and be consistent!

If you are the analyst, hopefully you were involved in the design stage so there will be fewer surprises

Data for one gene

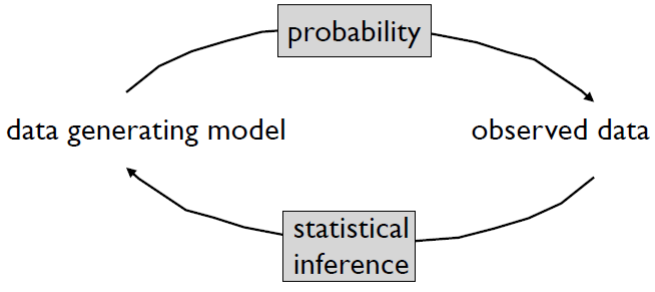
Why do the values vary?



Order of samples is arbitrary at this point

Making sense of the data

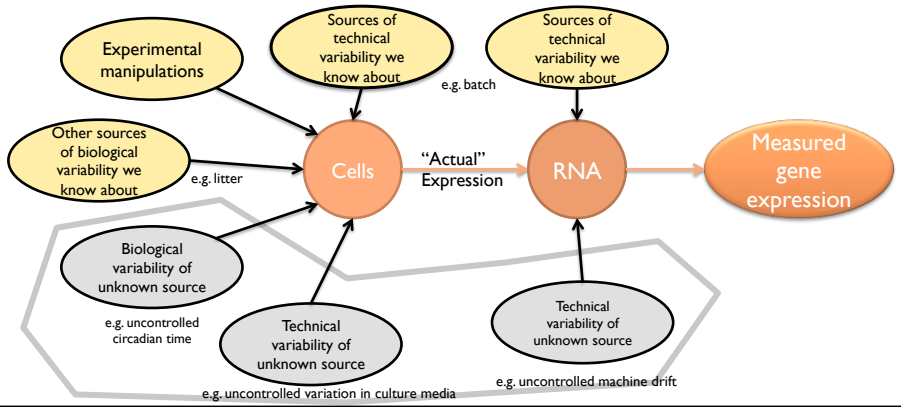
- Review: Data is what we observe, we want to infer something about "where it came from"



Jenny Bryan

Model of where expression data comes from

- The measured expression level of $gene_i$ is the combination of many effects.
- Analysis goal is often to determine relative role of effects, separate interesting from "uninteresting"
- One person's noise is another's signal.



Variability: friend and foe

- First line of defense: Know the enemy
  - You can only "correct" for things you know about.
  - Keep track of potential sources of variance: Batches of reagents, slides, personnel
- Design experiments to minimize impact of technical variability.
  - Avoid batches / minimize batch differences
  - Randomize design with respect to batches
- Replication
  - Biological (important)
  - Technical (*usually* less important but might need to convince yourself)
- Much more later in course

Exploratory numbers

Use as a rough description of the data

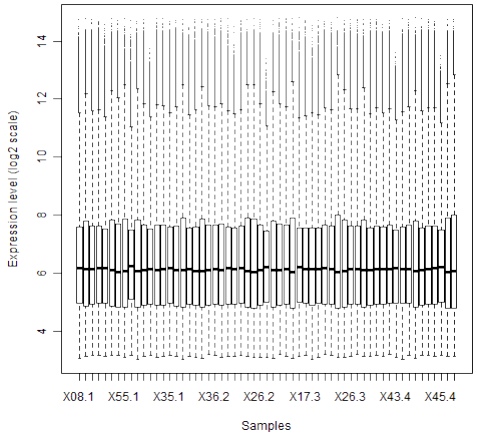
- Range
- Mean, Mode, Variance
- Number of missing values

Examine these values per-sample and per-element (e.g. gene)!

Exploratory graphics

- "Exploratory analysis" is "compute a little and graph a lot"
- I'll show a few simple approaches that are common in genomics
- Basis for examples: a data set of 59 Affymetrix microarrays, each of which have 54675 measurements. RNA was from blood cells of patients being treated for periodontal infection.

Boxplots to compare samples



Quick and dirty; Reasonable tool for summarizing large amounts of data.

Not so great if your distribution is multimodal.

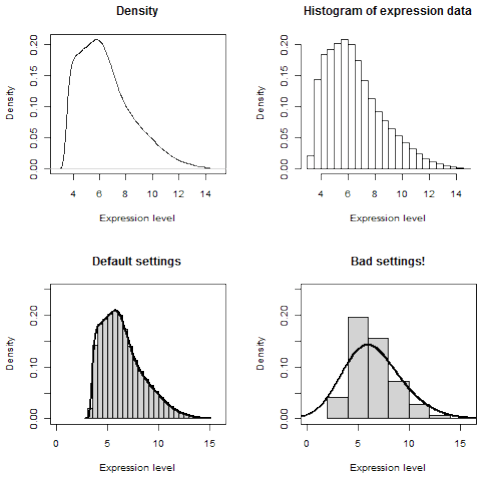
Don't use boxplots if you don't have to – show the actual data (esp. for small sample sizes!); use a strip plot

```
boxplot(dat, pch='.', xlab="Samples", ylab="Expression level (log2 scale)")
dev.print("basic.boxplot.png", device=png, width=500)
boxplot(dat[,1], pch='.', xlab=names(dat)[1], ylab="Expression level (log2 scale)")
```

Histograms

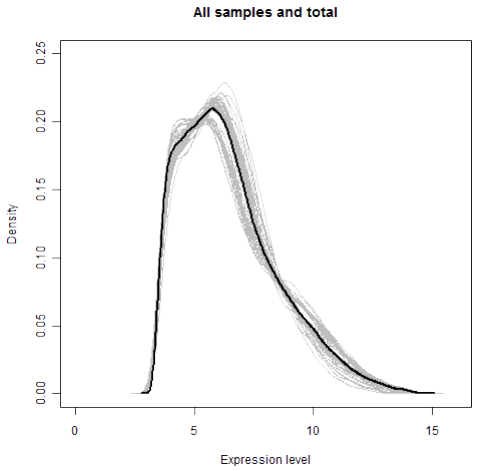
- Box plot's big brother but takes up more space.
- Consider using "density()" instead
- Choose sensible bin sizes and bandwidth

Histogram and density examples

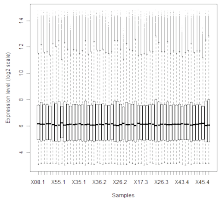



```
par(mfrow=c(2,2))
plot(density(as.matrix(dat)), xlab="Expression level", main="Density")
hist(as.matrix(dat), xlab="Expression level", main="Histogram of expression data", freq=F)
plot(0,xlim=c(0,16), ylim=c(0,0.25), xlab="Expression level", type="n", ylab="Density", main="Combined")
hist(as.matrix(dat), add=T, freq=F, col="lightgrey")
lines(density(as.matrix(dat)), xlab="Expression level", main="Density", lwd=2)
plot(0,xlim=c(0,16), ylim=c(0,0.25), xlab="Expression level", type="n", ylab="Density", main="Bad settings!")
hist(as.matrix(dat), add=T, freq=F, col="lightgrey", breaks=5)
lines(density(as.matrix(dat), bw=2), xlab="Expression level", main="Density", lwd=2)
```

Overlay densities of all samples



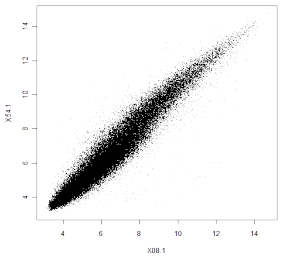
Contrast this with the boxplots.



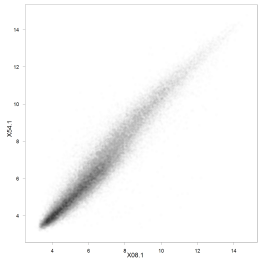
```
plot(0,xlim=c(0,16), ylim=c(0,0.25), xlab="Expression level", type="n", ylab="Density", main="All samples and total")
apply(dat, 2, function(x) { lines(density(x), col="grey") })
lines(density(as.matrix(dat)), xlab="Expression level",  lwd=2)
```

Scatter plots

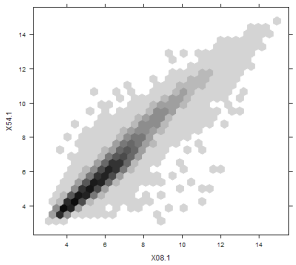
plot()



qplot(… alpha=I(0.01))



hexbinplot()

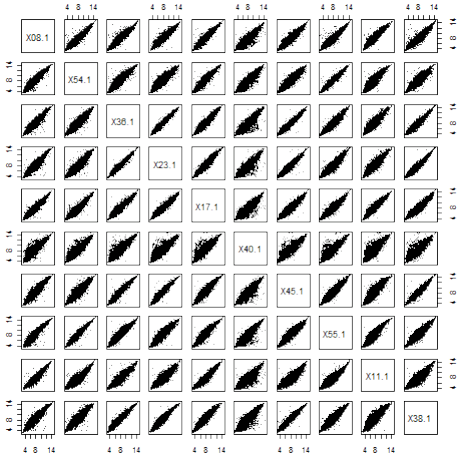



```
plot(dat[,1],dat[,2], xlab=names(dat)[1], ylab=names(dat)[2]  )
plot(dat[,1],dat[,2], xlab=names(dat)[1], ylab=names(dat)[2], pch='.' )
dev.print("basic.dotplot.png", device=png, width=500)
dev.off()

library(ggplot2)
p<-qplot(dat[,1],dat[,2], alpha=I(0.01),  xlab=names(dat)[1], ylab=names(dat)[2])
p+theme_bw()
last_plot() + opts(panel.grid.minor = theme_blank())
last_plot() + opts(panel.grid.major = theme_blank()) ggsave("fancy.dotplot.png")
ggsave("fancy.dotplot.ps") # doesn't work with alpha transparency

library(hexbin) # requires lattice
hexbinplot(dat[,1] ~ dat[,2], xlab=names(dat)[1], ylab=names(dat[2]), colorkey=F)
dev.print("hexbin.dotplot.png", device=png, width=500)
```

Alpha transparency looks cool but output is a bitmap, can't convert to postscript. Hexbin is a good alternative.

See also ggplot2 for more sophisticated options

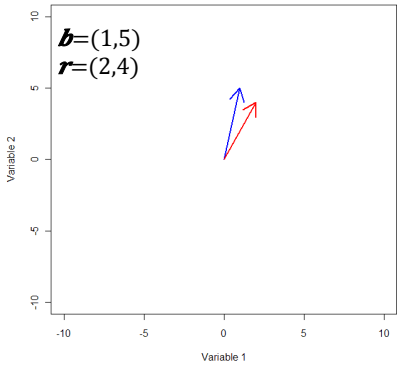
Comparing many pairs of samples



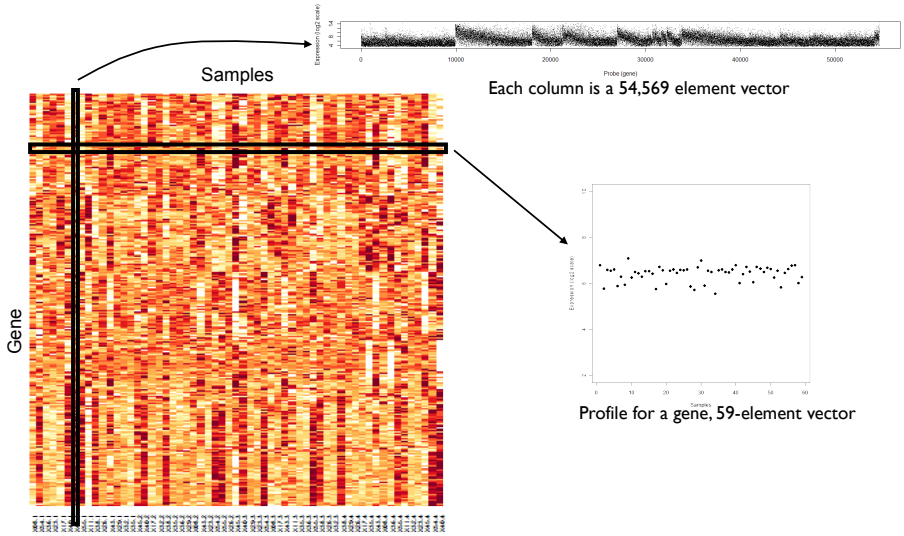

```
# plotting too much data:
pairs(dat, pch='.')
# get "Error in plot.new() : figure margins too large"
pairs(dat[,1:10], pch='.')
```

This is nice but unwieldy (or won't work) for large data sets

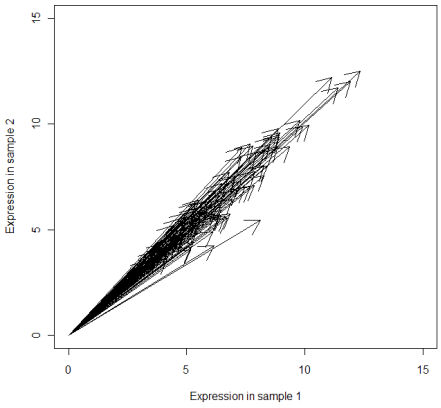
Digression into vectors



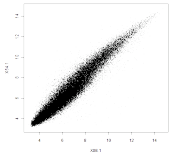
Rows and columns of your data matrix are vectors



First two dimensions of a bunch of gene vectors



Each vector is a gene (just 100 shown).
Variables/Coordinates are "assayed samples"

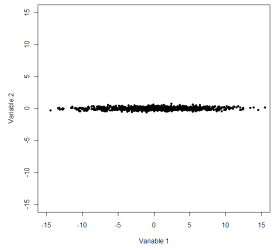


Without the arrows you can plot all the genes

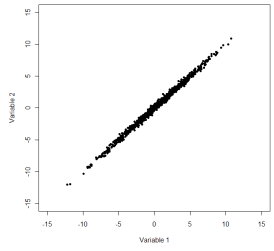
Do the genes form "interesting patterns" in "sample space"?

Data as points in space

- Distributed with a certain shape (and orientation.)
- Want to find "interesting patterns" in these "clouds"
- Do we need all 59 (or 54569) dimensions?
  - Often we are interested in finding interesting lower-dimensional structure in the (or representations of) data (regression, clustering, PCA)
- Motivating examples:



One of these dimensions is useless

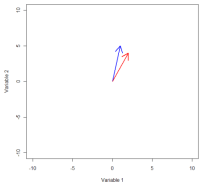


Dimensions are redundant

Comparing vectors: Euclidean distance

- The length of the difference between two vectors.
- In high dimensions, distances get big.

$$d(x, y) = \|x - y\| = \sqrt{\sum (x_i - y_i)^2}$$



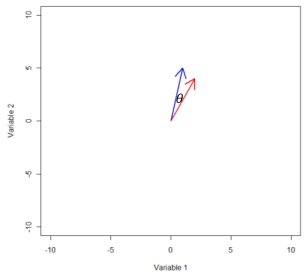
$\boldsymbol{b}=(1,5)$
$\boldsymbol{r}=(2,4)$
$\|\boldsymbol{r}-\boldsymbol{b}\|=?$

For matrices in R: `dist()`

Comparing vectors: Dot product

- Also known as **inner product**
- If two vectors are "nearer" each other, it's bigger
- Can be negative, unlike Euclidean distance

$$x \cdot y = \sum x_i y_i = \cos\theta \|x\| \|y\|$$



$\boldsymbol{b} = (1,5)$
$\boldsymbol{r} = (2,4)$
$\boldsymbol{b} \cdot \boldsymbol{r} = 2 + 20 = 22$

In R: %*%

Covariance

Recall: Sample variance $$s^2(\mathbf{x}) = \frac{\sum_i^N (x_i - \overline{x})^2}{N-1}$$

Sample covariance $$\mathrm{Cov}(\mathbf{x}, \mathbf{y}) = \frac{\sum_i^N (x_i - \overline{x})(y_i - \overline{y})}{N-1}$$

"Are you far from your mean when I am far from mine?"

What happens if the data are mean centered?

Comparing vectors: Pearson correlation

- Two vectors are pos. correlated if they point "in the same direction"
  - Equivalently, if the angle between them is small
- Always between 1 and -1, no matter how many dimensions.
- What happens if the data is **standardized** (mean 0, var 1)?

Covariance:

$$r(x, y) = \frac{\sum (x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum (x_i - \overline{x})^2 \sum (y_i - \overline{y})^2}}$$

Variances: "normalizes" the covariance

In R: `cov()` and `cor()`; `scale()` for standardizing

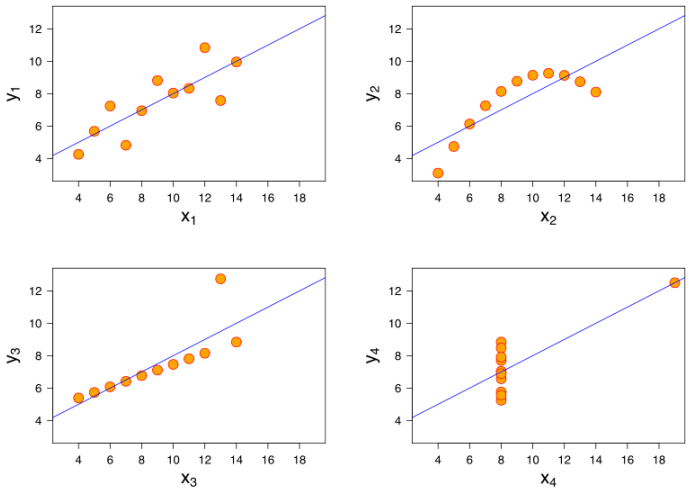
Self-study exercise: think about the metrics on the last few slides for cases of:

- orthogonal vectors
- vectors with opposite direction
- vectors with same direction
- High-dimensional data vs. lower-dimensional data
- What happens if variance of one vector is zero
- etc…

Supplement with sketches and/or plots in R

Look at the data!

Correlation does not imply "interesting", and lack of correlation does not mean "boring"

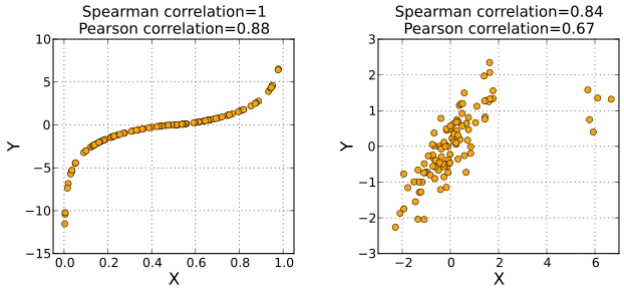


http://en.wikipedia.org/wiki/File:Anscombe%27s_quartet_3.svg

Remedy for some problems: ranks

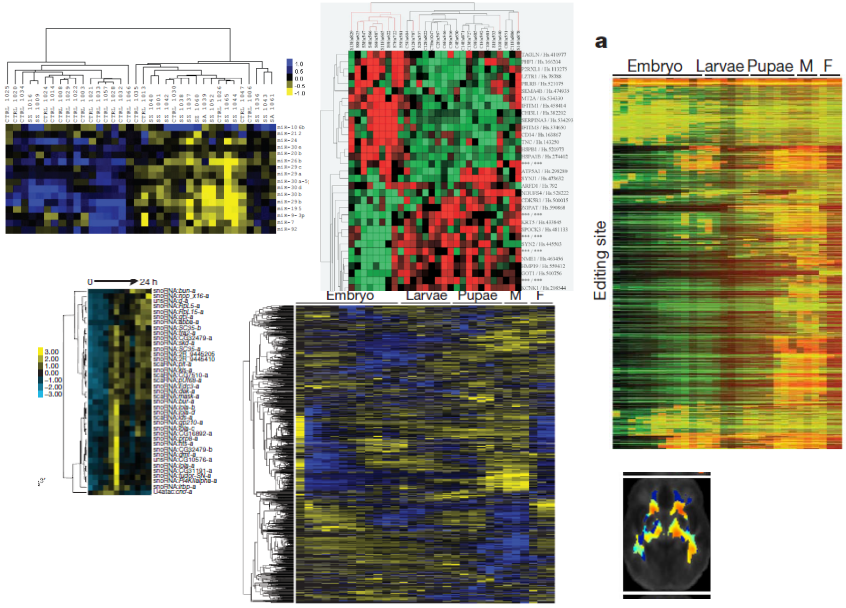
The correlation of the ranks is robust to "non-linear" relationships and outliers.

- Annoyance: dealing with ties



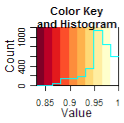
Wikipedia

Heat maps

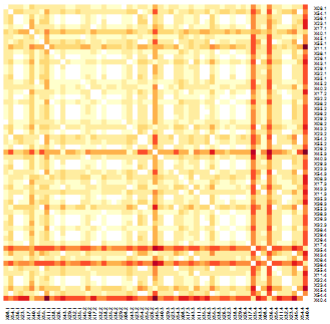


Using heat maps to compare assays

- Pairwise Pearson correlations of entire sample expression profiles

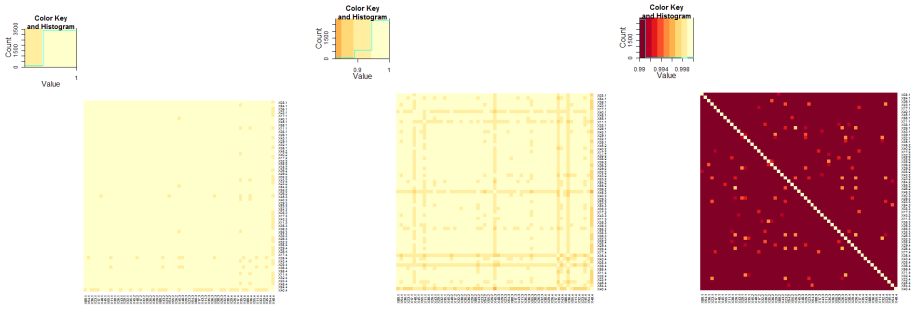


- This method loses some information, but is suitable for quite large data sets.
- Not for comparing many features (genes)!



```
library(gplots)
library(RColorBrewer)
cols<-c(rev(brewer.pal(9,"YlOrRd")), "#FFFFFF")
heatmap.2(cor(dat), Rowv=NA, Colv=NA, symm=T,
trace="none", dendrogram="none", col=cols, cexCol=0.5,
cexRow=0.5)
dev.print("heatmap.png", device=png, width=500)
```

Choose an appropriate range

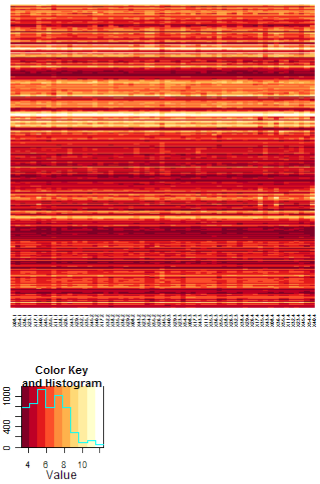


R sets "reasonable" ranges for you; but you can control it (`breaks`) and will often want to.
Depending on the distribution of values, might want to do `diag(m)<-NA` so the diagonal doesn't dominate the color space (say, if the correlations are all <<1)

Setting up data heat maps

- I've taken 500 rows at random from my big data set
- The heat map here was created using this data "as is".

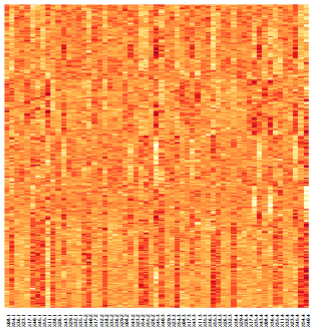
- I did cheat a little: The row ordering is not random.
- Note that this lecture doesn't get into dealing with missing data – how to evaluate it and visualize it



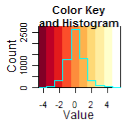

```
heatmap.2(bitOfData, Rowv=NA, Colv=NA, scale=NULL, trace="none", dendrogram="none", col=cols, cexCol=0.5, labRow=NA)
```

Revision 1

- Same input data, but each row is scaled to have mean 0 and variance 1 (z-scores)
  - Subtract the mean; divide by the standard deviation. use `scale()` on the data rows.
- It is now easier to compare the rows and seem some structure.
- But looks kind of bland compared to ones you often see in the literature



Note: R heatmap scales rows by default. You must disable this to give yourself control over how it turns out.



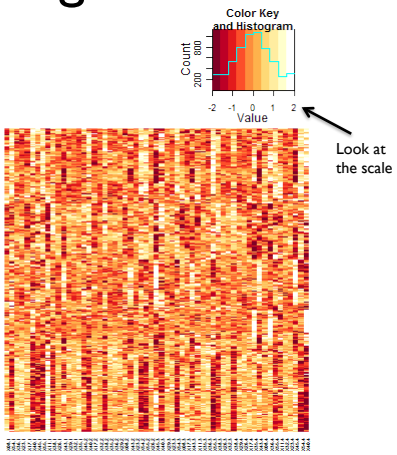

```
heatmap.2(scaledBit, Rowv=NA, Colv=NA, scale=NULL, trace="none", dendrogram="none", col=cols, cexCol=0.5,
cexRow=0.5, labRow=NA)
```

Revision 2: Adjusting contrast

- Same input data again, but after scaling I clipped the range of values to {-2,2}
- This means "anything more than two standard deviations from the row mean is set to 2"
- Now extremes have no effect; "Higher contrast"

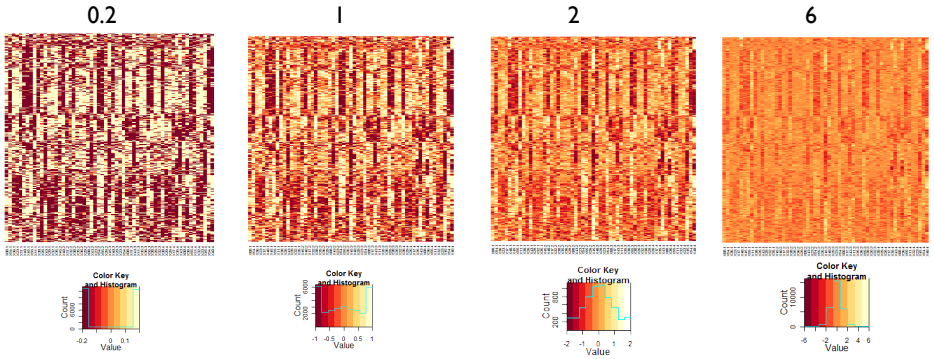
- Limit values of the range limit of 2 or 3 are usually good.

Heat maps you see in the literature are almost always set up this way. Be aware of what's really going on.



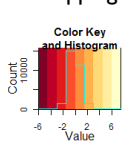

```
limit<-2
trim.scaledBit<-scaledBit
trim.scaledBit[which(trim.scaledBit < -limit)]<- -limit
trim.scaledBit[which(trim.scaledBit > limit)]<- limit
heatmap.2(trim.scaledBit, Rowv=NA, Colv=NA, scale=NULL, trace="none", dendrogram="none",
col=cols, cexCol=0.5, cexRow=0.5, labRow=NA)
```

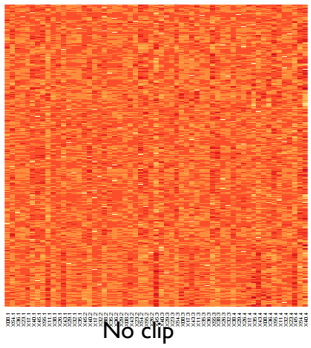
Varying clip limit



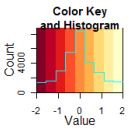
Another reason why clipping is and isn't good

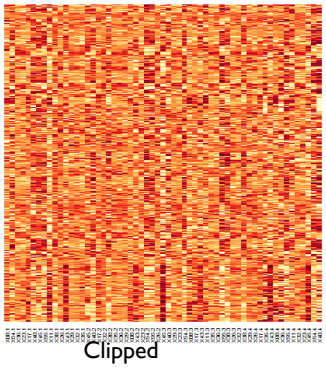
I added sparse random "spiky" values to the data (I just multiplied them by 3)
Clipping hides these "outliers" but allows us to see variation in the bulk of the data





No clip





Clipped

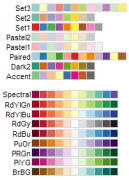
Plotting too much data

The entire data set.

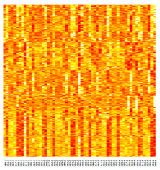
If the cells are less than 1 pixel, everything starts to turn to mush and can even be misleading.

(This won't work in R unless you print directly to a file.)

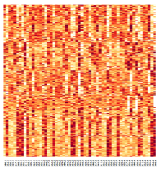
Choice of colours



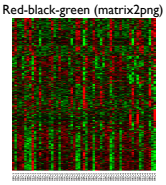
R default



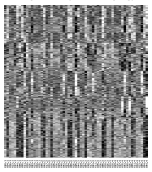
RColorBrewer YlOrRd



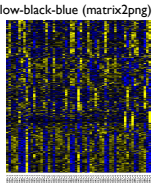
Red-black-green (matrix2png)



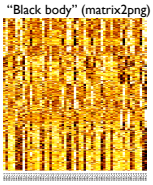
Greyscale (matrix2png)



Yellow-black-blue (matrix2png)

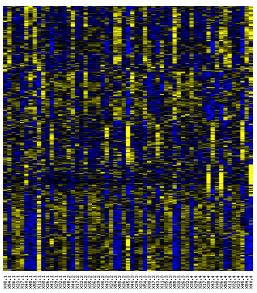


"Black body" (matrix2png)

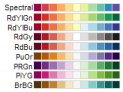


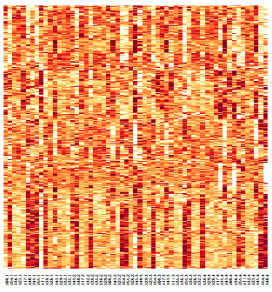
- R defaults: ketchup and mustard
- RColorBrewer: good maps based on work of visualization expert (matrix2png uses them too)
- Red-black-green you often see in papers is **bad choice** for colour blind individuals.
- Blue-black-yellow is better
- Greyscale: loss of dynamic range, but cheaper to publish!
- My favourite: matrix2png "black body"
- Humans can't really tell the difference between a 8 and 16 colour scale.

Divergent vs. sequential maps



Colours pass through black at (in this case) zero. Yellow="Below the mean", Blue="Above the mean" for the row. This can be the right thing to do, especially if your original data are naturally "symmetric" around zero (or some other value). Otherwise it might just be confusing.

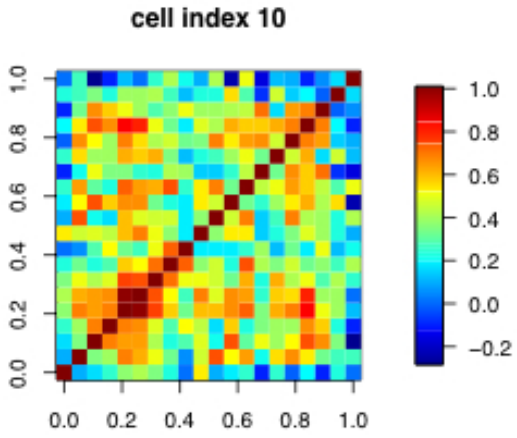




Colours go from light to dark. Darker colours mean "higher" by default in RColorBrewer; I usually reverse to make dark mean less.

A confusing heat map

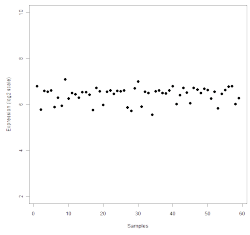
(at least for me; one can get used to anything)



http://mikelove.wordpress.com/category/visualization/

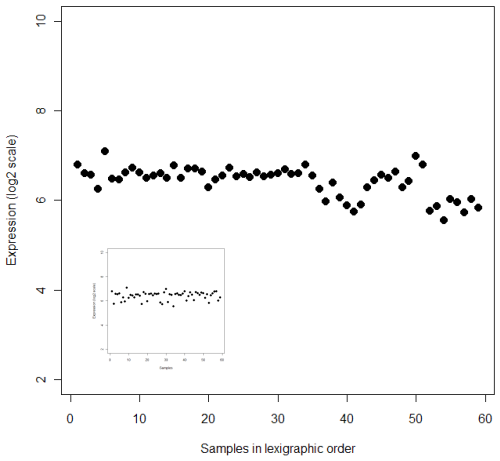
Slicing and dicing

- I mean: Viewing data arranged or grouped by factors of interest
- In my example, I have multiple "visits" for each "patient"
  - Any interesting trends we can spot?
  - Are intra-subject samples more similar than inter-subject samples?



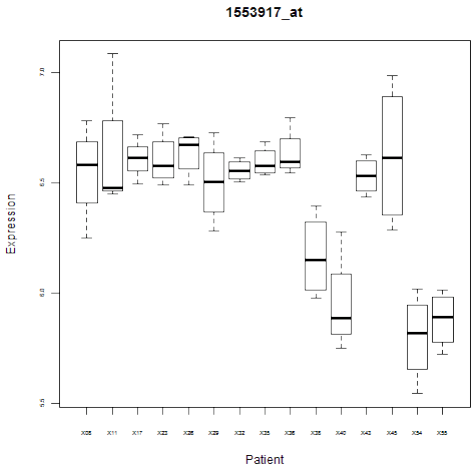
Playing around with one gene

- I sorted the samples by their names
- Is this structure meaningful? Is it interesting?
- Maybe we need more information.
- (And some statistics)



Explicitly group by patient

- We're developing the concept that we can look for "factors" that "explain" *some* of the variation.
- Is this interesting? We don't know, but it suggests that accounting for Patient differences might be reasonable.
- Need to get more comprehensive view …

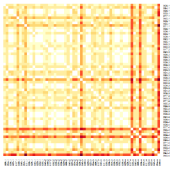


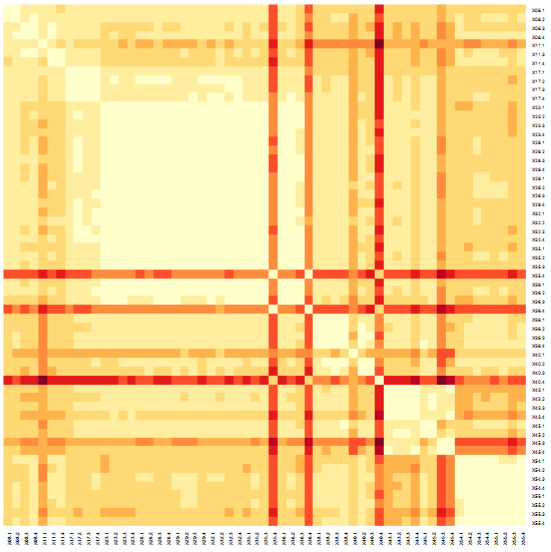
(Boxplots not ideal for this, but easy)

Arrange data by Patient

The sample correlations again; dark = low values

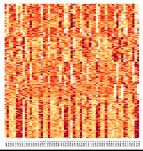
Original

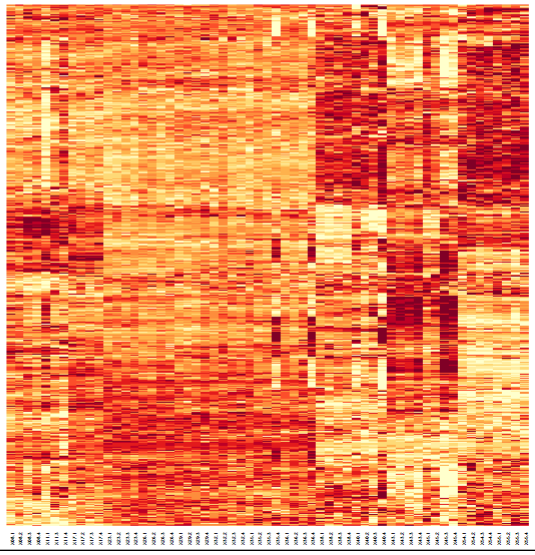




View of (part of) the data organized by Patient

(Again, row order is not random either)





Summary

- Sanity checks
- Use graphs to look at the data, slicing and dicing – build up a library of techniques that work for your data
- Additional exploratory techniques will be discussed later in the course
  - Clustering
  - PCA
- Also: missing values