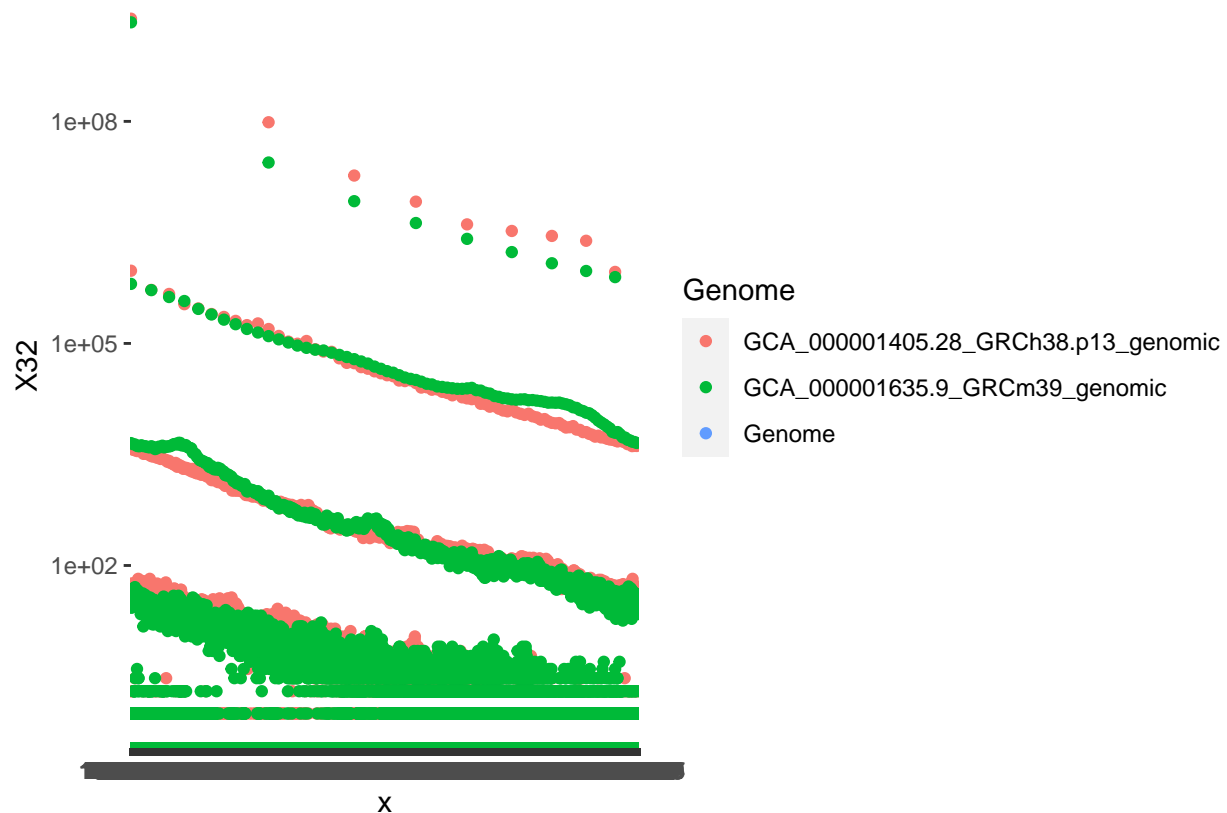


```
#4~k
#so as k increces

#human: 3billion
#mouse: about the same size, 3.1,2.8
#might want to be proportional at a later point
```

```
Dataframe%>%
  ggplot(aes(x=x,y=X32,color=Genome))+geom_point()+scale_y_log10()
```



```
longerVersion%>%
  filter(value!=0)
```

```
## # A tibble: 603,376 x 5
```

##	Genome	x	name	value	weighted
##	<chr>	<dbl>	<chr>	<dbl>	<dbl>
##	1 GCA_000001405.28_GRCh38.p13_genomic	1	X11	2638	2638
##	2 GCA_000001405.28_GRCh38.p13_genomic	1	X12	257757	257757
##	3 GCA_000001405.28_GRCh38.p13_genomic	1	X13	4267981	4267981
##	4 GCA_000001405.28_GRCh38.p13_genomic	1	X14	37984112	37984112
##	5 GCA_000001405.28_GRCh38.p13_genomic	1	X15	171077907	171077907
##	6 GCA_000001405.28_GRCh38.p13_genomic	1	X16	615097747	615097747
##	7 GCA_000001405.28_GRCh38.p13_genomic	1	X17	1263212344	1263212344
##	8 GCA_000001405.28_GRCh38.p13_genomic	1	X18	1737852949	1737852949

```
## 9 GCA_000001405.28_GRCh38.p13_genomic 1 X19 1988073244 1988073244
## 10 GCA_000001405.28_GRCh38.p13_genomic 1 X20 2110409768 2110409768
## # ... with 603,366 more rows
```

#8450895/9054271

```
numericnamesSumer%>%
  filter(name<20)%>%
  ggplot()+aes(x=name)+geom_point(aes(y=mean))+scale_y_log10()+geom_point(aes(y=random,color="3 billion /4^k"))
```

