task 1

In [2]:
```r
install.packages("ggplot2");install.packages("maps");install.packages("ggmap");install.packages("mapproj");
library(ggplot2); library(maps); library(ggmap); library(mapproj)
#installs and allows us to use nessecary packages


us <- map_data("state")
#sets us to data for states that is needed to map USA
us$state_name <- tolower(us$region)
#makes a new collumn that is the regions but lowercase so it can be used
map <- ggplot(us, aes(map_id= state_name))
#it makes the plot with data from us
map <- map + aes(x=long, y=lat, group=group) + geom_polygon()
#makes the axis latitude and logitude and makes it a shape not a line
map <- map + expand_limits(x=us$long, y=us$lat)
#sets the range of the axis based on USA
map <- map + coord_map() + ggtitle("USA Map")
#plots it based on the correct coordinates and names it
map
#calls it
```

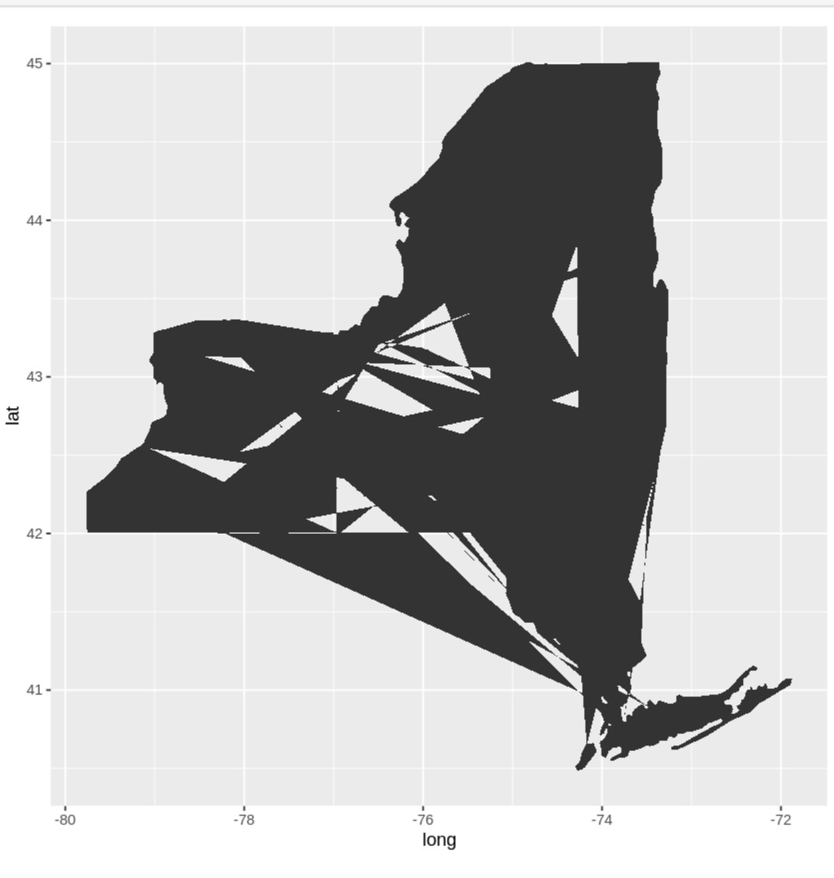
```
also installing the dependency ‘rlang’


Updating HTML index of packages in '.Library'

Making 'packages.html' ...
 done

Updating HTML index of packages in '.Library'

Making 'packages.html' ...
 done

also installing the dependencies ‘sp’, ‘RgoogleMaps’, ‘rjson’


Updating HTML index of packages in '.Library'

Making 'packages.html' ...
 done

Updating HTML index of packages in '.Library'

Making 'packages.html' ...
 done

Google's Terms of Service: https://cloud.google.com/maps-platform/terms/.

Please cite ggmap if you use it! See citation("ggmap") for details.
```

task 2

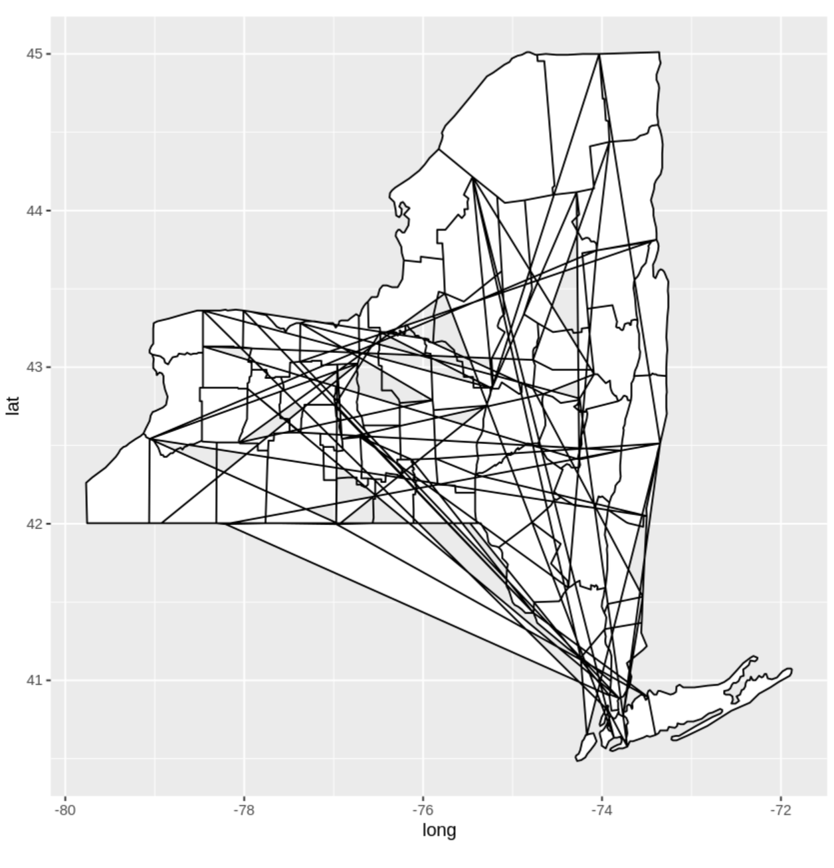
In [3]:
```r
map <- ggplot(us, aes(map_id= state_name))
map <- map + aes(x=long, y=lat, group=group) + geom_polygon(fill="white",color="black")
map <- map + expand_limits(x=us$long, y=us$lat)
map <- map + coord_map() + ggtitle("USA Map")
map
```

task 3

In [4]:
```r
ny_counties<-map_data("county","new york")
ggplot(ny_counties)+aes(long,lat)+geom_polygon()
```



In [5]:
```r
ggplot(ny_counties)+aes(long,lat)+geom_polygon(fill="white",color="black")
```



task 4

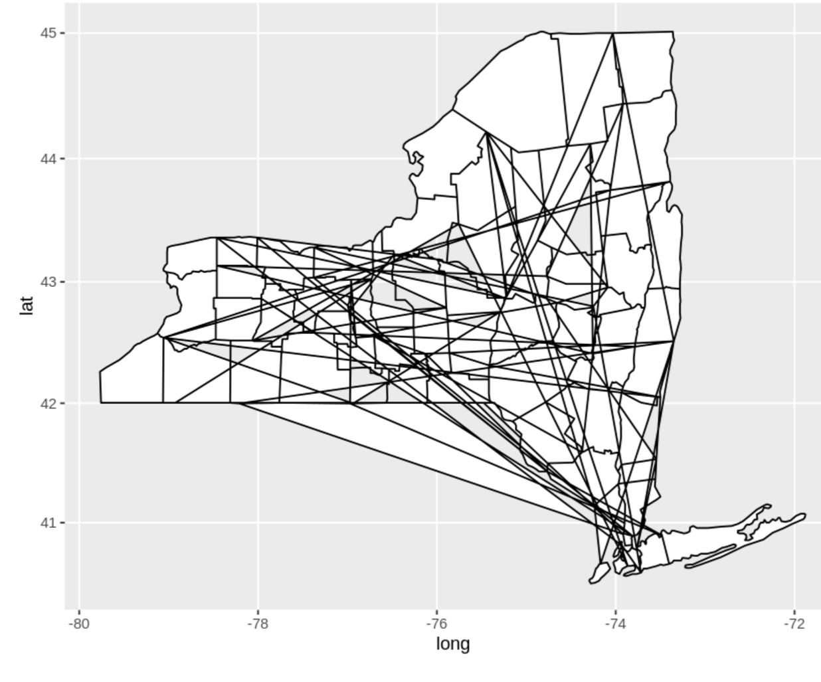
In [6]:
```r
head(ny_counties)
```

A data.frame: 6 × 6

	long	lat	group	order	region	subregion
	<dbl>	<dbl>	<dbl>	<int>	<chr>	<chr>
1	-73.78550	42.46763	1	1	new york	albany
2	-74.25533	42.41034	1	2	new york	albany
3	-74.25533	42.41034	1	3	new york	albany
4	-74.27252	42.41607	1	4	new york	albany
5	-74.24960	42.46763	1	5	new york	albany
6	-74.22668	42.50774	1	6	new york	albany

task 5

In [7]:
```r
ggplot(ny_counties)+aes(long,lat)+geom_polygon(fill="white",color="black")+coord_map(projection="mercator")
#slightly streached and distorted
```



task 6

In [11]:
```r
nyData<-data.frame(read.csv("https://ist387.s3.us-east-2.amazonaws.com/lab/nyData.csv"))
nyData
```

A data.frame: 62 × 5

county	pop2010	pop2000	sqMiles	popDen
<chr>	<chr>	<chr>	<chr>	<chr>
albany	304,204	294,565	522.8	581.87
allegany	48,946	49,927	1,029.31	47.55
bronx	1,385,108	1,332,650	42.1	32,900.43
broome	200,600	200,536	705.77	284.23
cattaraugus	80,317	83,955	1,308.35	61.39
cayuga	80,026	81,963	691.58	115.71
chautauqua	134,905	139,750	1,060.23	127.24
chemung	88,830	91,070	407.35	218.07
chenango	50,477	51,401	893.55	56.49
clinton	82,128	79,894	1,037.85	79.13
columbia	63,096	63,094	634.71	99.41
cortland	49,336	48,599	498.76	98.92
delaware	47,980	48,055	1,442.44	33.26
dutchess	297,488	280,150	795.63	373.9
erie	919,040	950,265	1,042.69	881.41
essex	39,370	38,851	1,794.23	21.94
franklin	51,599	51,134	1,629.12	31.67
fulton	55,531	55,073	495.47	112.08
genesee	60,079	60,370	492.94	121.88
greene	49,221	48,195	647.16	76.06
hamilton	4,836	5,379	1,717.37	2.82
herkimer	64,519	64,427	1,411.47	45.71
jefferson	116,229	111,738	1,268.59	91.62
kings	2,504,700	2,465,326	70.82	35,367.13
lewis	27,087	26,944	1,274.68	21.25
livingston	65,393	64,328	631.76	103.51
madison	73,442	69,441	654.84	112.15
monroe	744,344	735,343	657.21	1,132.58
montgomery	50,219	49,708	403.04	124.6
nassau	1,339,532	1,334,544	284.72	4,704.73
⋮	⋮	⋮	⋮	⋮
oneida	234,878	235,469	1,212.43	193.72
onondaga	467,026	458,336	778.39	599.99
ontario	107,931	100,224	644.07	167.58
orange	372,813	341,367	811.69	459.3
orleans	42,883	44,171	391.26	109.6
oswego	122,109	122,377	951.65	128.31
otsego	62,259	61,676	1,001.70	62.15
putnam	99,710	95,745	230.31	432.94
queens	2,230,722	2,229,379	108.53	20,553.97
rensselaer	159,429	152,538	652.43	244.36
richmond	468,730	443,728	58.37	8,030.32
rockland	311,687	286,753	173.55	1,795.95
saratoga	219,607	200,635	809.98	271.13
schenectady	154,727	146,555	204.52	756.54
schoharie	32,749	31,582	621.82	52.67
schuyler	18,343	19,224	328.33	55.87
seneca	35,251	33,342	323.71	108.9
st lawrence	111,944	111,931	2,680.38	41.76
steuben	98,990	98,726	1,390.56	71.19
suffolk	1,493,350	1,419,369	912.05	1,637.36
sullivan	77,547	73,966	968.13	80.1
tioga	51,125	51,784	518.6	98.58
tompkins	101,564	96,501	474.65	213.98
ulster	182,493	177,749	1,124.24	162.33
warren	65,707	63,303	866.95	75.79
washington	63,216	61,042	831.18	76.06
wayne	93,772	93,765	603.83	155.3
westchester	949,113	923,459	430.5	2,204.68
wyoming	42,155	43,424	592.75	71.12
yates	25,348	24,621	338.14	74.96

task 7

In [13]:
```r
mergeNY<-merge(ny_counties,nyData,all.x=TRUE,by.x="subregion",by.y="county")
```

task 8

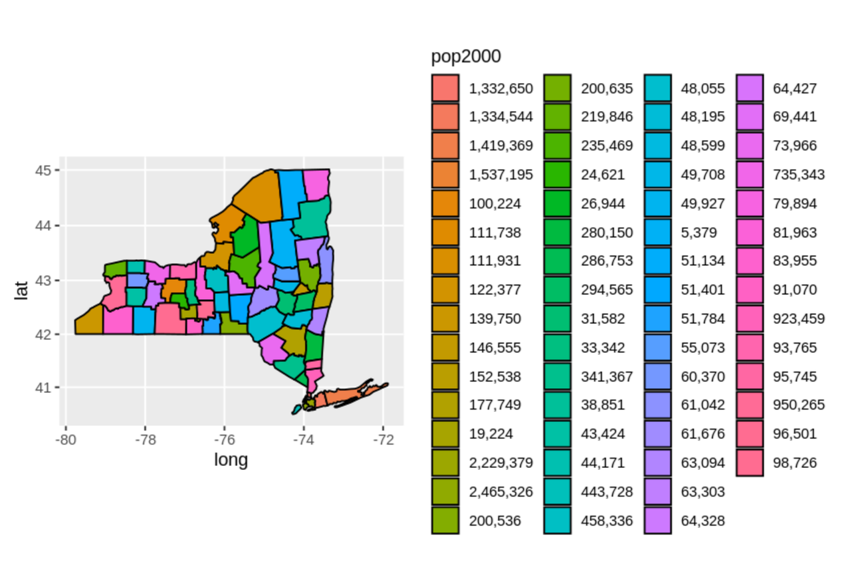
In [14]:
```r
head(mergeNY)
```

A data.frame: 6 × 10

	subregion	long	lat	group	order	region	pop2010	pop2000	sqMiles	popDen
	<chr>	<dbl>	<dbl>	<dbl>	<int>	<chr>	<chr>	<chr>	<chr>	<chr>
1	albany	-73.78550	42.46763	1	1	new york	304,204	294,565	522.8	581.87
2	albany	-74.25533	42.41034	1	2	new york	304,204	294,565	522.8	581.87
3	albany	-74.25533	42.41034	1	3	new york	304,204	294,565	522.8	581.87
4	albany	-74.27252	42.41607	1	4	new york	304,204	294,565	522.8	581.87
5	albany	-74.24960	42.46763	1	5	new york	304,204	294,565	522.8	581.87
6	albany	-74.22668	42.50774	1	6	new york	304,204	294,565	522.8	581.87

task 9

In [16]:
```r
ggplot(mergeNY)+aes(long,lat)+geom_polygon(aes(fill=pop2000),color="black")+coord_map(projection="mercator")
```



task 10

I'm not going to do this stop because it's optional, and I have a lot of work to do today