# coalescent_simulation_report

March 22, 2016

## 1 Analysis of the Coalescent Simulation

```
In [1]: library(ggplot2)
        library(plyr)
        library(reshape2)
```

### 1.1 Dataset

This is the summary of Spearman's $\rho$ over 10 replicates of the "coalescent" experiment

```
In [2]: stats = read.csv("overall.csv")
        stats$rep = as.factor(sort(rep(1:10, times=28)))
```

```
In [3]: summary(stats)
```

```
Out[3]:     coverage     measure       scale           spearman            rep
         Min.   : 0.50   ip :140   Min.   :0.001   Min.   :0.3145   1      : 28
         1st Qu.: 4.00   wip:140   1st Qu.:0.010   1st Qu.:0.7894   2      : 28
         Median :22.50             Median :0.010   Median :0.8641   3      : 28
         Mean   :19.46             Mean   :0.019   Mean   :0.8353   4      : 28
         3rd Qu.:30.00             3rd Qu.:0.010   3rd Qu.:0.9159   5      : 28
         Max.   :50.00             Max.   :0.100   Max.   :0.9727   6      : 28
                                                                    (Other):112
```

We compare average genome coverage and the scale of varaition againsnt accuracy (i.e. Spearman's $\rho$) (over the 10 reps).

We compare the effects of coverage and scale independently.

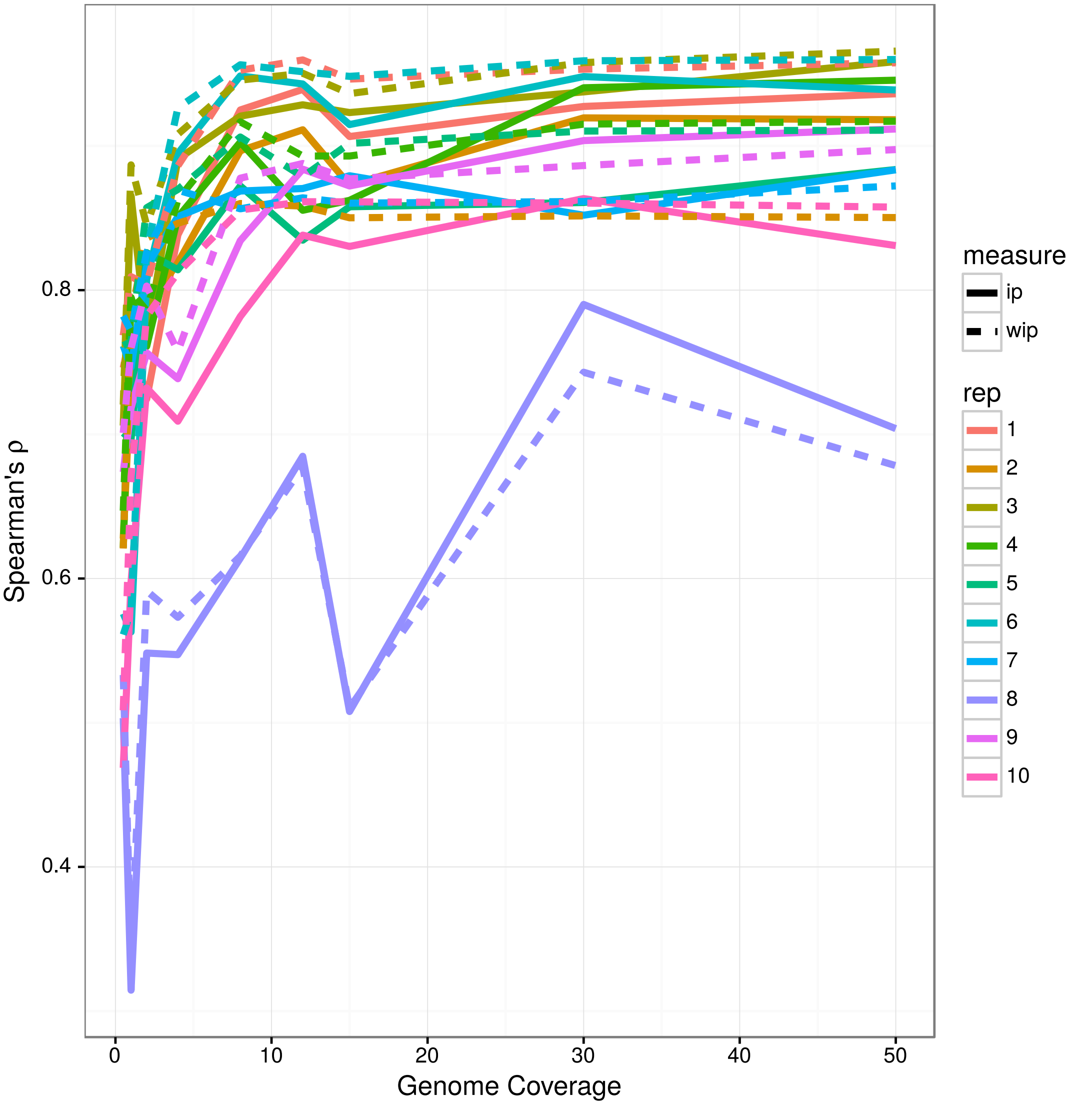
### 1.2 Coverage vs $\rho$

A series of average coverages was run at the scale of 0.01 (i.e. an average of 1 variant in 100 bases across all pairwise comparisions of samples)

```
In [4]: coverage = stats[stats$scale==0.01, ]
        coverage$scale = NULL
        summary(coverage)
```

```
Out[4]:     coverage     measure     spearman            rep
         Min.   : 0.50   ip :90    Min.   :0.3145   1      :18
         1st Qu.: 2.00   wip:90    1st Qu.:0.7592   2      :18
         Median : 8.00             Median :0.8555   3      :18
         Mean   :13.61             Mean   :0.8118   4      :18
         3rd Qu.:15.00             3rd Qu.:0.9029   5      :18
         Max.   :50.00             Max.   :0.9658   6      :18
                                                    (Other):72
```

```
In [5]: ggplot(coverage, aes(x=coverage, y=spearman, linetype=measure, color=rep)) +
            geom_line(aes(linetype=measure, color=rep),size=1.5) +
            xlab('Genome Coverage') +
            ylab(expression(paste("Spearman's ", rho))) +
            #scale_x_log10()+
            theme_bw()
```



Here we summarise the replicates to averages ± SD. Note that we exclude replicate 8 as it is an outlier for both IP and WIP metrics (see above).

```
In [6]: csumm = ddply(coverage, .(coverage, measure), summarise,
                      spearman_m=mean(spearman),
                      spearman_sd=sd(spearman))
        summary(csumm)
```

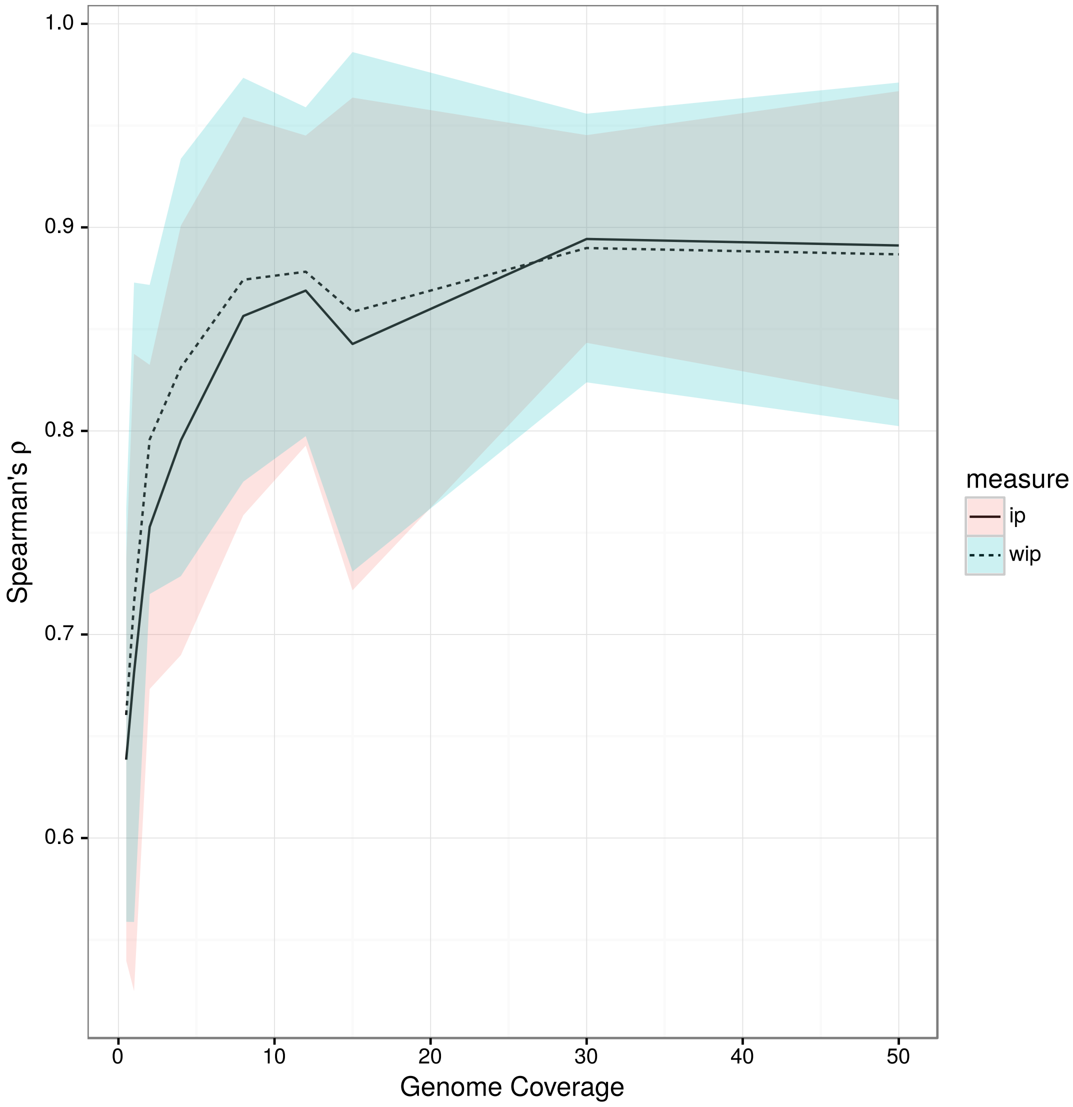
```
Out[6]:     coverage      measure   spearman_m       spearman_sd
         Min.   : 0.50   ip :9   Min.   :0.6385   Min.   :0.05102
```

```
 1st Qu.: 2.00   wip:9   1st Qu.:0.7635   1st Qu.:0.07703
 Median : 8.00           Median :0.8496   Median :0.09837
 Mean   :13.61           Mean   :0.8118   Mean   :0.09764
 3rd Qu.:15.00           3rd Qu.:0.8772   3rd Qu.:0.10476
 Max.   :50.00           Max.   :0.8943   Max.   :0.15706
```

You can see below that WIP marginally outperforms IP, at low coverage. Above about 20x, I would say that WIP and IP have equivalent performance.

The ribbon is 1 SD, so there is certainly no signficant difference.

```
In [7]: ggplot(csumm, aes(x=coverage, y=spearman_m, ymin=spearman_m-spearman_sd, ymax=spearman_m+spearma
            geom_line(aes(linetype=measure)) +
            geom_ribbon(aes(fill=measure), alpha=0.2) +
            xlab('Genome Coverage') +
            ylab(expression(paste("Spearman's ", rho))) +
            #scale_x_log10()+
            theme_bw()
```



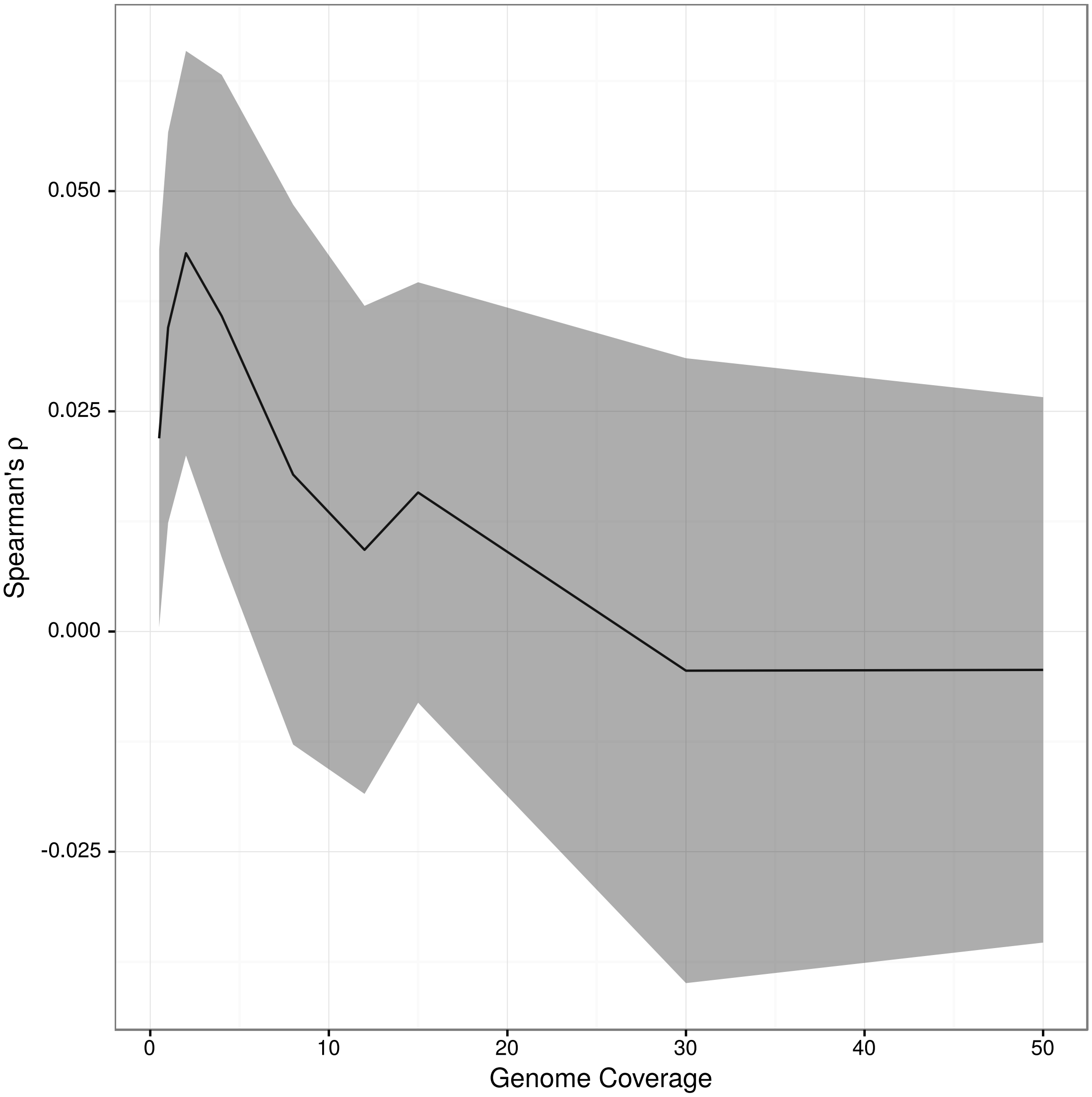
The differnce between WIP and IP is calculated here

```
In [8]: cdiff = dcast(coverage, coverage * rep˜ measure, value.var="spearman")
        #cdiff = ddply(cdiff, .(coverage, rep), summarise, spearman_d=wip - ip)
        cdiff = ddply(cdiff, .(coverage), summarise, diff_m=mean(wip - ip), diff_sd=sd(wip - ip))

        summary(cdiff)
```

```
Out[8]:     coverage          diff_m              diff_sd
         Min.   : 0.50   Min.   :-0.004446   Min.   :0.02145
         1st Qu.: 2.00   1st Qu.: 0.009273   1st Qu.:0.02297
         Median : 8.00   Median : 0.017816   Median :0.02738
         Mean   :13.61   Mean   : 0.018810   Mean   :0.02696
         3rd Qu.:15.00   3rd Qu.: 0.034508   3rd Qu.:0.03065
         Max.   :50.00   Max.   : 0.042950   Max.   :0.03547
```

```
In [9]: ggplot(cdiff, aes(x=coverage, y=diff_m, ymin=diff_m-diff_sd, ymax=diff_m+diff_sd)) +
            geom_line() +
            geom_ribbon(alpha=0.4) +
            xlab('Genome Coverage') +
            ylab(expression(paste("Spearman's ", rho))) +
            #scale_x_log10()+
            theme_bw()
```



### 1.3 Scale vs $\rho$

Like coverage, we investigate the effect of variation at a constant coverage, in this case 30x. I also convert the scale into its inverse, as this is how some people prefer to think of it (i.e. one variant in X bases, as opposed to 0.0x variants per base on average. Each to their own. . . )

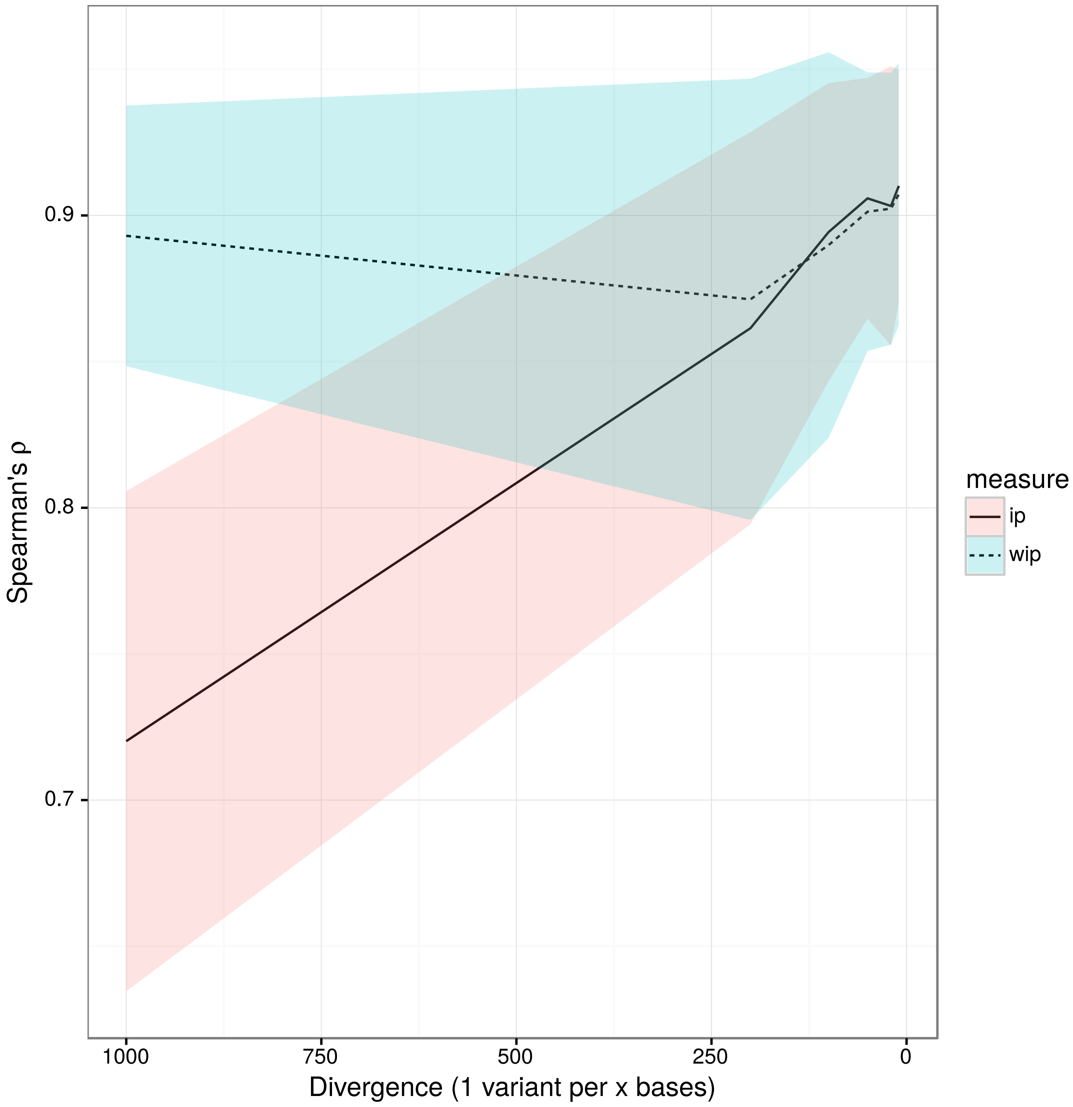
```
In [10]: scale = stats[stats$coverage==30, ]
         scale$scale = 1/scale$scale
```

```
In [11]: ssumm = ddply(scale, .(scale, measure), summarise,
                       spearman_m=mean(spearman),
                       spearman_sd=sd(spearman))
         summary(csumm)
```

```
Out[11]:     coverage     measure   spearman_m      spearman_sd
          Min.   : 0.50   ip :9   Min.   :0.6385   Min.   :0.05102
```

```
 1st Qu.: 2.00   wip:9   1st Qu.:0.7635   1st Qu.:0.07703
 Median : 8.00           Median :0.8496   Median :0.09837
 Mean   :13.61           Mean   :0.8118   Mean   :0.09764
 3rd Qu.:15.00           3rd Qu.:0.8772   3rd Qu.:0.10476
 Max.   :50.00           Max.   :0.8943   Max.   :0.15706
```

```
In [12]: ggplot(ssumm, aes(x=scale,  y=spearman_m, ymin=spearman_m-spearman_sd, ymax=spearman_m+spearmar
             geom_line(aes(linetype=measure)) +
             geom_ribbon(aes(fill=measure), alpha=0.2) +
             xlab('Divergence (1 variant per x bases)') +
             ylab(expression(paste("Spearman's ", rho))) +
             #scale_x_log10() +
             scale_x_reverse() +
             theme_bw()
```



### 1.4 Conclusions

- I think there might be an issue with the way I normalise trees. I think that we are probably at a higher level of divergence than I expect if we use the mean. I will do a run with a couple of reps using the maximum distance set to 1.0, i.e. that the entire tree scale is 0.5 (from root to tip, and then back again =1.0).
- I'd like to re-do the coverage sweep at a scale of 0.005 or 0.002 or even 0.001. I think that this might be more inline with our rice experiment. My take home from this is that WIP is only important when your signal:noise ratio is low, like when you have a small amount of variation. Otherwise, they are equivalent (neither is signficantly worse on average). Norman, can you comment?

```
In [ ]:
```