Phenome imputation of a panel of prototrophic haploid yeast segregants

Hannah Meyer

2018-05-29

Contents

The following vignette outlines the strategy for the imputation of missing phenotype values in 1,008 prototrophic haploid yeast segregants that were phenotyped for 46 traits [1]. Before imputation, the missing data patterns are investigated for missing data mechanisms. The find the best imputation strategy, the subset of fully phenotyped segregants is chosen and artifical missingness introduced. A number of different imputation techniques (phenix [2], mice [3], mvn [4] [2]) are then used to impute these masked phenotypes. By comparing these results to the originally measured phenotypes, the most suitable imputation method for the dataset can be determined.

1. Dataset

The dataset generated by [1] consists of phenotype and genotype data of 1,008 prototrophic haploid *Saccharomyces cerevisiae* segregants derived from a cross between a laboratory strain (BY MATa) and a wine strain strain (RM MATα). In brief, the segregants were generated by mating of the haploid parental strains and subsequent sporulation of the diploid heterozygote. Sporulation resulted in 1,008 four-spore tetrads that showed 2:2 segregation of mating type and drug-resistance markers. From each tetrad one spore was selected for further analyses. For phenotyping, these segregants were grown on agar plates in 46 growth conditions. These can broadly be grouped into growth on different carbohydrates or derivatives thereof (lactose, lactate, raffinose, maltose, mannose, sorbitol, trehalose, xylose, galactose), growth on different culture media (YPD, YNB) with different pH (YNB:pH3, YNB:pH8) or in different temperatures (YPD:4C, YPD:15C, YPD:37C), growth on different antibiotics and xenotbiotics (e.g. cadmium chloride, neomycin, zeocin, cis platin, for a full list, see labels in figure 4. After incubation for 48h, the colony size of each segregant grown in the different conditions was measured. The final phenotypes were defined as the colony size normalised to colony size growth on control medium.

Segregants were genotyped using Illumina short-read sequencing. After mappping, quality control and filtering for unique genotype markers, all 1,008 segregants were genotyped 11,623 unique genotypic markers.

The phenix uses the infitesimal genetic contribution –estimated as sample-by-sample relatedness matrix– to impute missing values. A relatedness matrix for the yeast segregants can be estimated on the sequenced genotypic markers. In brief, the genetic markers were pruned for markers that are in LD within a window of 3kb and show a correlation $r^2 > 0.2$. As the dataset is based on an F2 cross, LD structure estimation is not straight-forward and this window size is a simple estimate derived from a study on the population genomics of domestic and wild yeasts [5]. The LD pruning reduced the marker set for kinship estimation to 4,105 SNPs. The kinship was estimated using the method introduced by [6]. PLINK [7] was used for both LD pruning (with parameters –indep-pairwise 3kb 5 0.) and kinship estimation (with parameters –make-rel square gz).

Original phenotype data from [1] and the relatedness matrix based on the genotypes:

```r
## libraries ####
library(ggplot2)

## directories ####
rawdir <- "~/data/LiMMBo/yeast/rawdata"
directory <- "~/data/LiMMBo/yeast/processeddata"

## data ####
geno <- read.table(paste(rawdir, "/BYxRM_GenoData.txt", sep=""), header=TRUE)
load(paste(rawdir, "/cross.Rdata", sep=""))
pheno <- cross$pheno
rownames(pheno) <- colnames(geno)[-1]
genoinfo <- cross$geno
kinship <- read.table(paste(rawdir, "/BYxRM.3kb.grm.rel.csv", sep=""), sep=",",
                      header=TRUE)

## general parameters ####
cutoff <- 0.95
col <- c('#fc8d62','#8da0cb')
text_size <- 12
```

2. Pattern of missing data

Figure 1 shows an aggregation plot (middle) where all existing combinations of missing (blue) and non-missing (orange) values in the traits are depicted. The bar chart on its right shows the frequencies of occurrence of the different combinations. The histogram on the top shows the frequency of missing values for each trait.

```r
plot_pattern_missingness(pheno, directory=directory,
                         name="missing_data_pattern_allTraits")
```

```r
frequency_missingness <- data.frame(missing=
                                        apply(pheno, 2, function(x)
                                            length(which(is.na(x)))/length(x)))
frequency_missingness$complete <- 1 - frequency_missingness$missing

per_sample_missingness <- data.frame(missing=
                                         apply(pheno, 1, function(x)
                                             length(which(is.na(x)))/length(x)))
per_sample_missingness$complete <- 1 - per_sample_missingness$missing

Samples2Keep <- per_sample_missingness$missing <= 0.20

pheno <- pheno[Samples2Keep,]
```

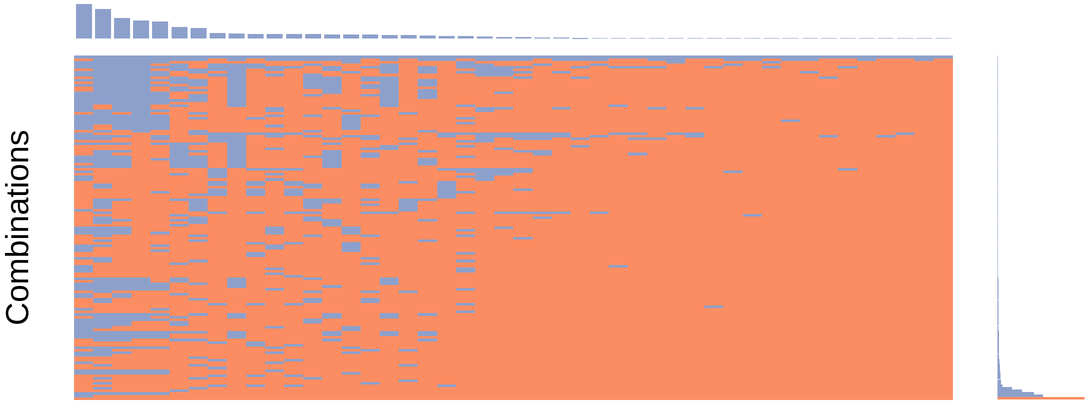


Figure 1: Frequencies and distributions of missing values in the complete yeast dataset with 1,008 segregants and 46 traits.

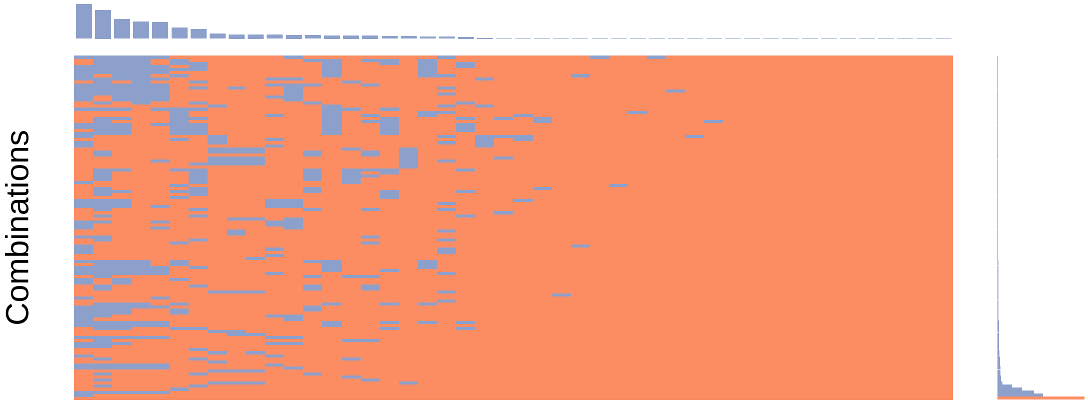


Figure 2: Frequencies and distributions of missing values in the yeast dataset filtered for samples missing more than 20% of traits.

```
kinship <- kinship[Samples2Keep, Samples2Keep]

plot_pattern_missingness(pheno, directory=directory,
                         name="missing_data_pattern_filteredTraits")
```

Any sample with more than 20% missing phenotype data is removed from further analyses, reducing the dataset to 981 samples. The aggregation and frequency plots for this reduced dataset is shown in figure 2.

3. Missing data mechanism

In general, one can distinguish between three missing data types: missing completely at random (MCAR), missing at random (MAR) and missing not at random (MNAR) [8]. For instance, consider an example where there are N colonies of yeast and one wants to automatically detect the size and the density of each colony with a suitable instrument. If the instrument fails with a constant probability for any colony independent of the measurement, then the pattern of missing values in the data is MCAR. If the probability that the density measurement is missing changes with the value of the size measurement, but is not dependent on the density of colonies with the same size, then the data are MAR. In contrast, data are MNAR if the probability of obtaining a density measurement depends on the density of colonies with the same size. In practice, detecting the missing data mechanism often proves difficult. Testing for MCAR can be done via statistical tests [4], but distinguishing between MAR and MNAR cannot be achieved formally as this would require knowledge of the missing values [8]. However, there are visualisation tools that provide diagnostic plots and approxim- ate measures which can help make assumptions about the missingness mechanism.

In the following, I use an implementation of Little's test for MCAR from the BaylorEdPsych R package.

```
MCARtest <- BaylorEdPsych::LittleMCAR(pheno)
#> this could take a while
pMCAR <- pchisq(MCARtest$chi.square, MCARtest$df, lower.tail=FALSE)
```

Little's method tests the null hypothesis that the data is MCAR [4], which can in this case be rejected with a p-value of 2.39416155676756e-34 (based on a χ^2 distribution, $\chi^2 = 5901.6260025$ with 4631 degrees of freedom).

Determining if data is MAR or MNAR cannot be tested for formally and relies on approximate measures and assumptions based on the experimental procedures. If it can be demonstrated that one or more variables in the dataset are significantly correlated with missing values, missingness may be predictable, which is the requirement for imputing MAR data. In order to test for predictable missingness, I created an indicator matrix for the phenotype matrix, where observed values were encoded as zero and missing values as one. For each of the 46 traits in the dataset, I correlated the observed values across all samples with each column of the indicator matrix, i.e. the missingness patterns per trait. If all values were observed for a given trait, all values in the indicator matrix in this column were equal to zero and the correlation between the trait and the missingness was set to NA. Figure 3 shows the correlation patterns between the phenotypes and the missing values per trait. For traits like cobalt chloride and magnesium sulfate, where little data is missing, many entries are NA. Overall, missingness of the phenotypes seems to be predictable.

```
corrMiss <- correlationMissingness(pheno)
plot_correlation_missingness(data=corrMiss, savePdf=TRUE,directory=directory,
                             addgrid.col="grey", labelsize=0.5)
```

4. Dataset with no missing values

```
pheno_noNA <- fullPhenotypes(pheno)

## a) correlation between phenotypes
corrPhenotypes <- correlationPhenotypes(pheno)

plot_correlation_phenotypes(data_r=corrPhenotypes$r,
                            data_p=corrPhenotypes$padjust, savePdf=TRUE,
                            directory=directory,
                            addgrid.col="grey", labelsize=0.5)
```

Out of the 981 segregant for which at least 80% of phenotypes were measured, 303 were fully phenotyped. The pairwise phenotype correlation across these segregants is shown in figure 4.

5. Generate dataset with artificial missingness

Imputation of missing values requires an understanding of which missing trait values can be reliably imputed. In order to do this, I needed a fully phenotyped dataset with the same structure as the yeast dataset, where missing values could be introduced, imputed and subsequently compared to the true values. I chose a simple approach using the subset of the 303 fully phenotyped samples and introducing missing values with a similar pattern of missingness as observed in the original dataset. The results for the real (figure 2) and simulated (5) dataset are similar in terms of frequencies and combinations of missing/non-missing traits.

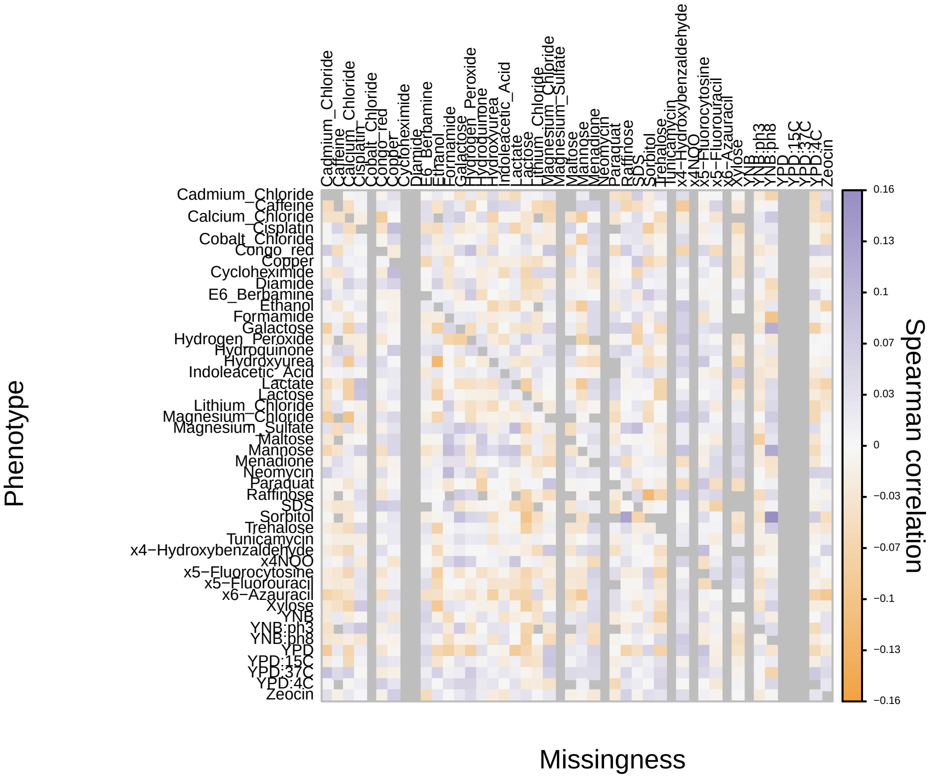


Figure 3: Correlations of observed phenotypes with missing data values.

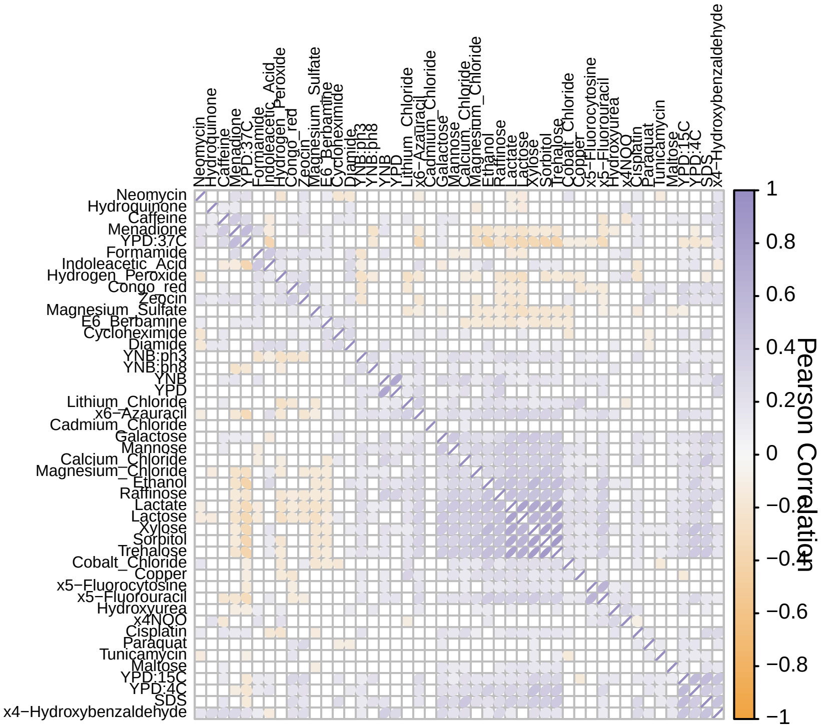


Figure 4: Pairwise correlations of 46 growth traits in yeast.

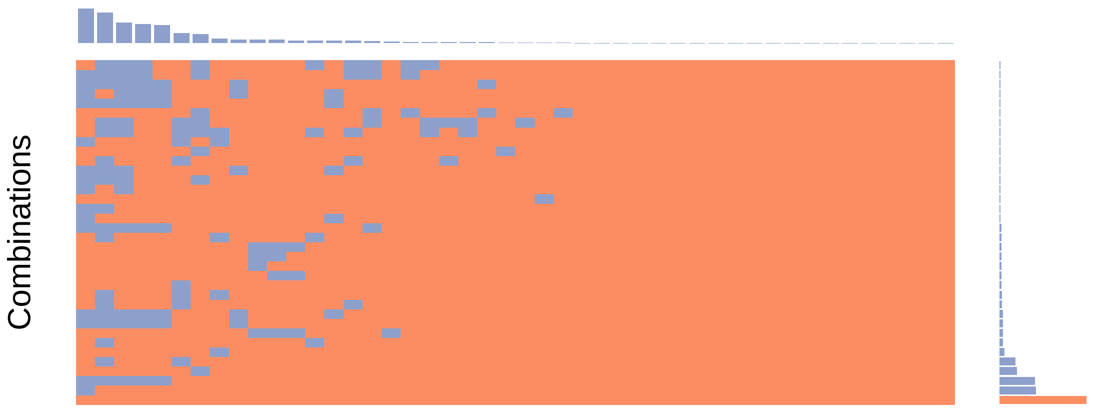


Figure 5: Frequencies and distributions of missing values for the fully-phenotyped dataset of 306 segregants with simulated missing values based on the observed missingness pattern for the pool of 991 segregants.

```
pheno_artificial <- artificialMissingness(data=pheno, fulldata=pheno_noNA,
                                          kinship=kinship, seed=3422)
plot_pattern_missingness(pheno_artificial$data_addNA, directory=directory,
                            name="missing_data_pattern_simulated", savePdf=TRUE)
```

6. Impute artifically created missing data

The masked values in the dataset with artificial missingness were imputed with one genetic (phenix) and two non-genetic (mvn and mice) methods. For imputation with mice, different predictor sets were analyses (All phenotypes as predictors, phenotypes with correlation > 0.1, 0.2 or greater 0.3), based on the correlation of the measured phenotypes (figure 4). After imputation, I evaluated the goodness of the imputation by computing the Pearson correlation of the imputed values to the experimentally observed ones (figure 6). Table 1 show those correlations across all traits for all imputation methods.

```
imputed <- imputeData(data=as.matrix(pheno_artificial$data_addNA),
                        fulldata=pheno_noNA,
                        kinship=pheno_artificial$kinship,
                        method=c("phenix", "mvn", "mice"),
                        cutoff=cutoff, testing=TRUE)
plot_overview_correlation_imputation(imputed$cor, savePdf=TRUE,
                                          directory=directory,
                                          text_size=8)
```

```
knitr::kable(imputed$summary,
caption="\\label{tab:imputation}Correlation of imputed vs real phenotype values")
```

Table 1: Correlation of imputed vs real phenotype values

	median	mean	sd
mice_0.1	0.9991321	0.9856103	0.0316208
mice_0.2	0.9992921	0.9842400	0.0352196
mice_0.3	0.9993773	0.9831504	0.0372470
mice_All	0.9995736	0.9857059	0.0327623
mvn	0.9992442	0.9876369	0.0256959
phenix	0.9995626	0.9872657	0.0271280

Imputation results per method and trait are shown below. Traits where the imputed values correlated to the original ones by less than 95% are marked in red.

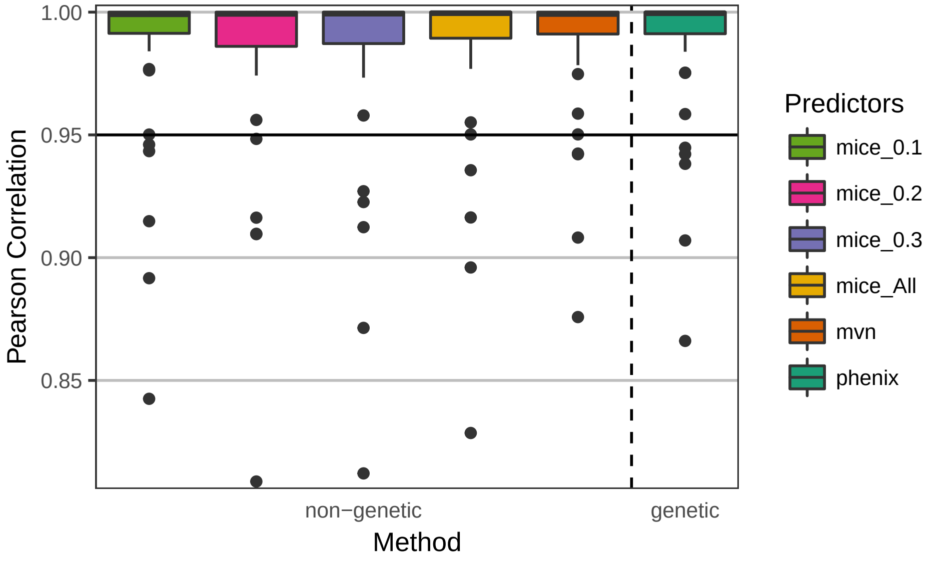
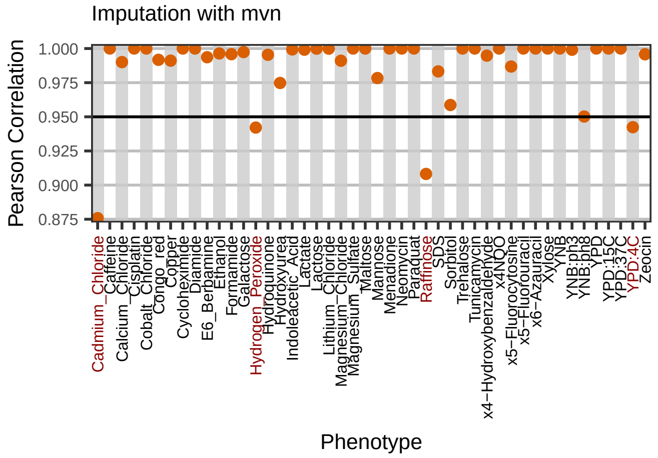
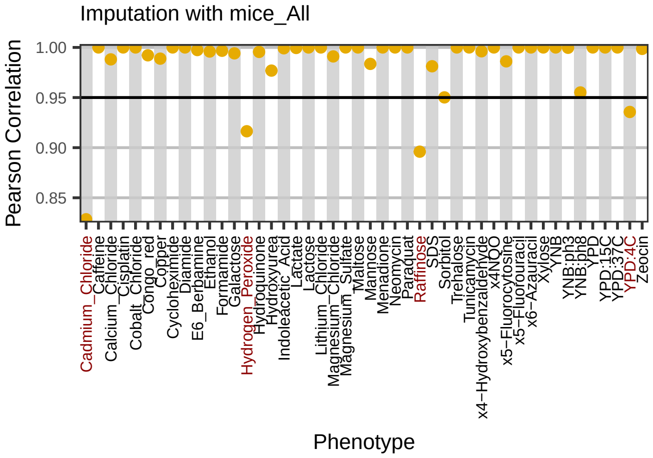
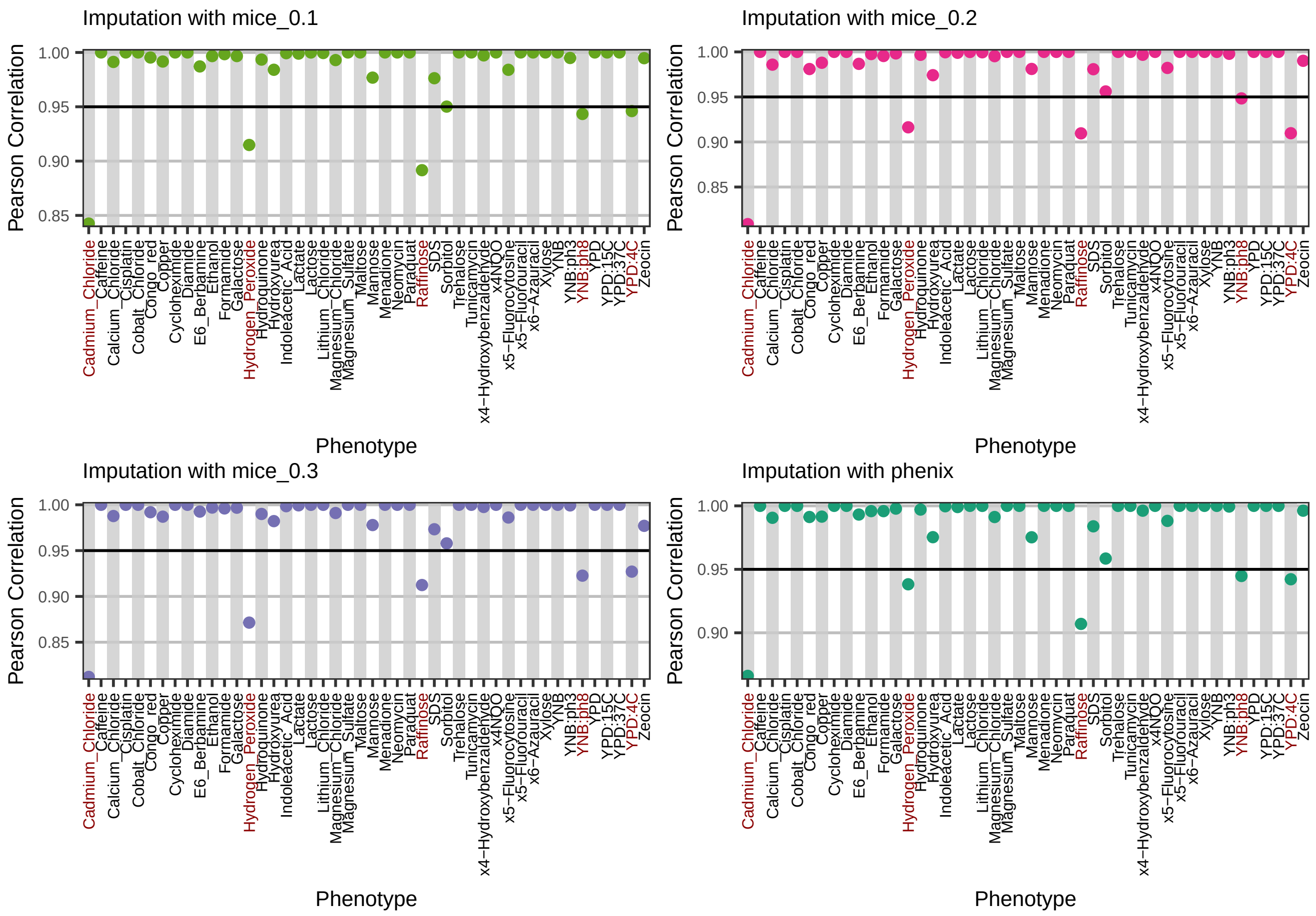


Figure 6: Pearson correlation of imputed values to the experimentally observed ones.

```r
corr_plots_methods <- plot_individual_correlation_imputation(imputed$cor,
                                                         savePdf=TRUE,
                                             directory=directory, text_size=6)
dummy <- lapply(corr_plots_methods, function(p) print(p))
```







7. Impute full data set

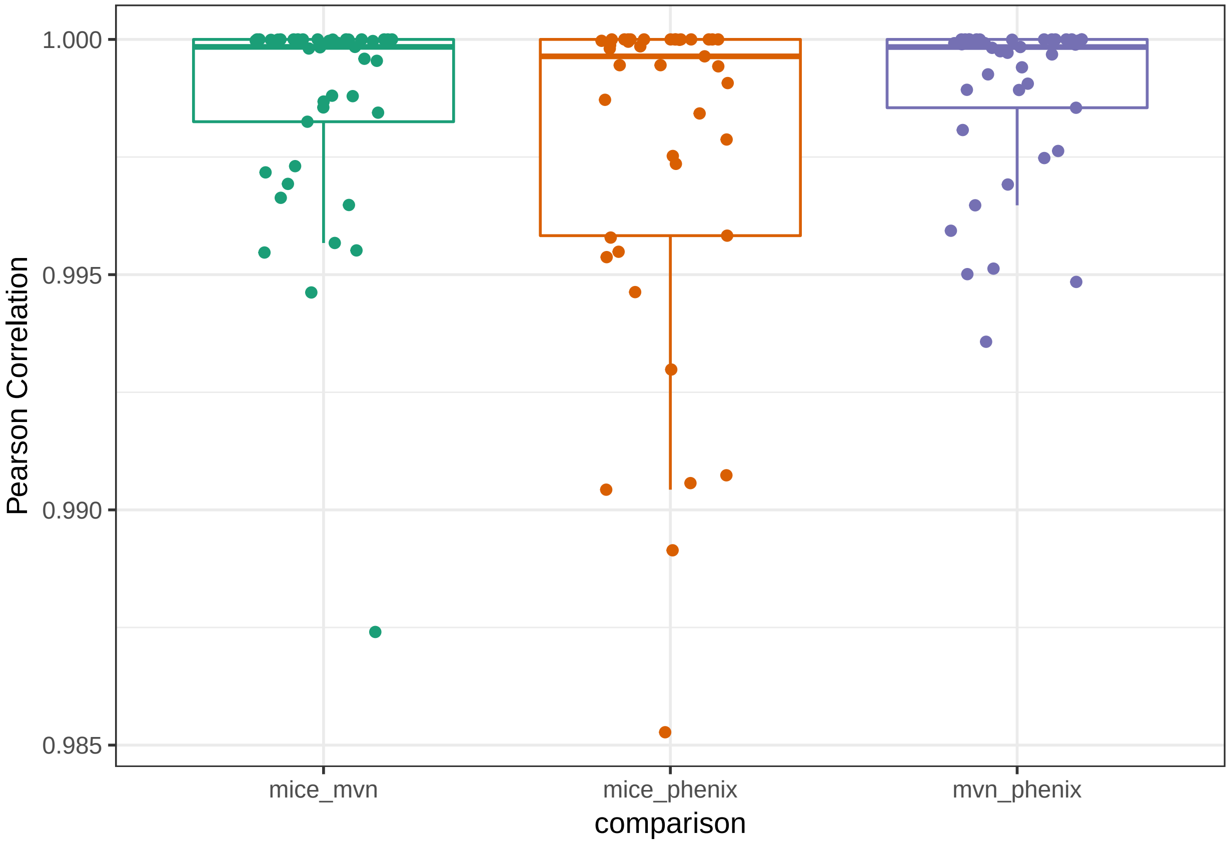
Traits that can be reliable imputed (correlation greater than 95%) are selected and the missing values from the phenotype set with 981 segregrants imputed.

```
imputable <-  data.frame(sapply(c("phenix", "mvn", "mice"), imputableTraits,
                                  imputed, cutoff))

imputed_phenos <- imputeData(data=pheno, methods= c("phenix", "mice", "mvn"),
                       imputable=imputable, cutoff=0.95, testing=FALSE,
                       kinship=as.matrix(kinship))
saveRDS(imputed_phenos, paste(directory, "/imputed_phenotypes.rds", sep=""))

pheno_phenix <- imputed_phenos$imp$phenix$imp
pheno_mvn <- imputed_phenos$imp$mvn$imp
pheno_mice <- imputed_phenos$imp$mice$imp

write.table(imputed_phenos$imp$phenix$imp, paste(directory,
                                 "/phenotypes_phenix.csv", sep=""),
            sep=",", col.names=NA, row.names=TRUE, quote=FALSE)

write.table(imputed_phenos$imp$mvn$imp, paste(directory,
                                 "/phenotypes_mvn.csv", sep=""),
            sep=",", col.names=NA, row.names=TRUE, quote=FALSE)

write.table(imputed_phenos$imp$mice$imp, paste(directory,
                                 "/phenotypes_mice.csv", sep=""),
            sep=",", col.names=NA, row.names=TRUE, quote=FALSE)
```

```
commonImputable <- intersect(intersect(colnames(pheno_mice), colnames(pheno_mvn)),
                             colnames(pheno_phenix))

mice_mvn <- do.call(rbind, lapply(commonImputable, function(x) {
    tmp <- Hmisc::rcorr(pheno_mice[,colnames(pheno_mice) == x],
                        pheno_mvn[,colnames(pheno_mvn) == x])
    return(data.frame(p=tmp$P[1,2], r2=tmp$r[1,2], comparison="mice_mvn", trait=x))
}))

mice_phenix <- do.call(rbind, lapply(commonImputable, function(x) {
    tmp <- Hmisc::rcorr(pheno_mice[,colnames(pheno_mice) == x],
                        pheno_phenix[,colnames(pheno_phenix) == x])
    return(data.frame(p=tmp$P[1,2], r2=tmp$r[1,2], comparison="mice_phenix", trait=x))
}))

phenix_mvn <- do.call(rbind, lapply(commonImputable, function(x) {
    tmp <- Hmisc::rcorr(pheno_phenix[,colnames(pheno_phenix) == x],
                        pheno_mvn[,colnames(pheno_mvn) == x])
    return(data.frame(p=tmp$P[1,2], r2=tmp$r[1,2], comparison="mvn_phenix", trait=x))
}))

compare_corr <- rbind(mice_mvn, mice_phenix, phenix_mvn)
compare_corr$comparison <- as.factor(compare_corr$comparison)
```

The pairwise correlation of the phenotypes imputed with mice, mvn and phenix is shown below. There is strong correlation bewtween the imputed phenotypes from all three methods, with the lowest correlation of 0.9852735 for Sorbitol in the comparison between mice and phenix imputation.

```
p <- ggplot(data=compare_corr, aes(x=comparison, y=r2,
                                   color=comparison))
p + geom_boxplot(outlier.colour = NA) +
    geom_label(data=dplyr::filter(compare_corr, r2 < 0.95), aes(y=r2,
                                                                x=comparison,
                                                                label=trait),
               nudge_y = -0.002, size=3) +
    geom_jitter(width = 0.2) +
    scale_color_manual(values=c('#1b9e77','#d95f02','#7570b3'), guide=FALSE) +
    ylab("Pearson Correlation") +
    theme_bw()
```


Pearson Correlation
1.000
0.995
0.990
0.985
mice_mvn
mice_phenix
mvn_phenix
comparison

8. References

1. Bloom JS, Ehrenreich IM, Loo WT, Lite T-LV, Kruglyak L. Finding the sources of missing heritability in a yeast cross. Nature. 2013;494: 234–7. doi:10.1038/nature11867

2. Dahl A, Iotchkova V, Baud A, Johansson Å, Gyllensten U, Soranzo N, et al. A multiple-phenotype imputation method for genetic studies. Nature Genetics. Nature Publishing Group; 2016;48: 466–472. doi:10.1038/ng.3513

3. Buuren S van, Groothuis-Oudshoorn K. mice : Multivariate Imputation by Chained Equations in R. Journal of Statistical Software. 2011;45: 1–67. doi:10.18637/jss.v045.i03

4. Little RJA. A Test of Missing Completely at Random for Multivariate Data with Missing Values. Source Journal of the American Statistical Association. 1988;83: 1198–1202. Available: http://www.jstor.org/stable/2290157 http://about.jstor.org/terms

5. Liti G, Carter DM, Moses AM, Warringer J, Parts L, James SA, et al. Population genomics of domestic and wild yeasts. Nature. Nature Publishing Group; 2009;458: 337. doi:10.1038/NATURE07743

6. Yang J, Lee SH, Goddard MEM, Visscher PMP, Hindorff L, Sethupathy P, et al. GCTA: a tool for genome-wide complex trait analysis. The American Journal of Human Genetics. Elsevier; 2011;88: 76–82. doi:10.1016/j.ajhg.2010.11.011

7. Chang CC, Chow CC, Tellier LC, Vattikuti S, Purcell SM, Lee JJ. Second-generation PLINK: rising to the challenge of larger and richer datasets. GigaScience. 2015;4: 7. doi:10.1186/s13742-015-0047-8

8. Little RJA, Rubin DB. Statistical analysis with missing data. 2nd ed. Balding DJ, Bloomfield P, Cressie NAC, Fisher NI, Johnstone IM, Kadane JB, et al., editors. New Jersey: John Wiley & Sons, Inc; 2002. p. 408.