Setting up docker on server

Stopping and starting all running containers

```
docker stop $(docker ps -a -q)
docker rm $(docker ps -a -q)
```

Pulling and running my pre-built images

The RStudio container has been altered to make rstudio a sudo user. Postgis has been installed in order to get shp2psql and raster2psql. Libgdal-dev has been added along with unixodbc. PLR has been added to the postgis image.

The flags set the data valumes to folders in the home directory.

```
## Note the data directories are given completely free permsions. No security here! However none is rea

mkdir -p /home/duncan/postgres_data
chmod 777 -R /home/duncan/postgres_data
mkdir -p /home/duncan/rstudio_server
chmod 777 -R /home/duncan/rstudio_server

docker run --name "postgis-plr" -p 25432:5432 -d -v /home/duncan/postgres_data:/var/lib/postgresql dgoli

docker run --name "rstudio" --link postgis-plr:postgis -d -p 8788:8787 -v /home/duncan/rstudio_server:/
```

R runs from http://172.16.49.31:8788

The default postgis user is docker with pwd docker

Using .pgpass

In order to run commands within R studio without being asked for a password for the default user (docker) with password docker needs a simple .pgpass file to be placed in the home directory of the user making the calls.

```
hostname:port:database:username:password

i.e

postgis:*:*:docker:docker
```

Save this as simply .pgpass The permissions need to be set to u=rw (0600) or less so go into the shell and change this with sudo chmod 0600 .pgpass

Creating data bases from R

```
nm<-"brant"
com<-paste("dropdb -h postgis -U docker ",nm,sep="")
com
```

```
## [1] "dropdb -h postgis -U docker brant"
```

```
system(com)
com<-paste("createdb -h postgis -U docker ",nm,sep="")
com
```

```
## [1] "createdb -h postgis -U docker brant"
```

```
system(com)
```

```
library(RODBC)
con<-odbcConnect("brant")
odbcQuery(con,"create extension postgis")
```

```
## [1] -1
```

```
odbcQuery(con,"create extension plr")
```

```
## [1] -1
```

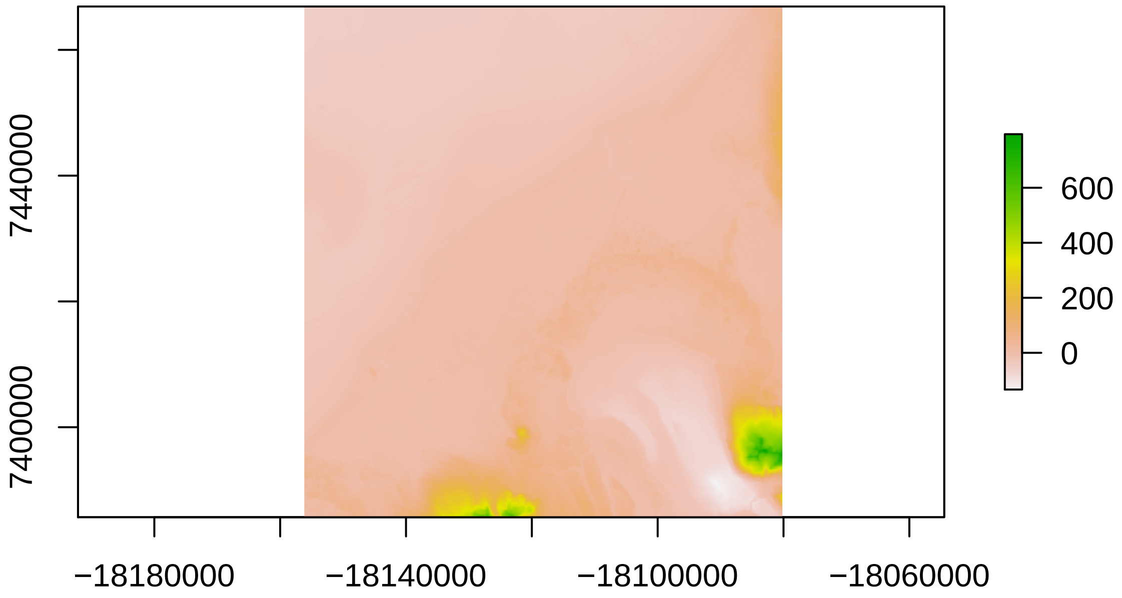
```
library(raster)
```

```
## Loading required package: sp
```

```
library(rgdal)
```

```
## rgdal: version: 1.2-4, (SVN revision 643)
##  Geospatial Data Abstraction Library extensions to R successfully loaded
##  Loaded GDAL runtime: GDAL 1.10.1, released 2013/08/26
##  Path to GDAL shared files: /usr/share/gdal/1.10
##  Loaded PROJ.4 runtime: Rel. 4.8.0, 6 March 2012, [PJ_VERSION: 480]
##  Path to PROJ.4 shared files: (autodetected)
##  Linking to sp version: 1.2-3
```

```
r<-raster("shiny_morph/cold_bay_3857_clip.tiff")
plot(r)
```



```
r_nm<-r@file@name
r_nm
```

```
## [1] "/home/rstudio/shiny_morph/cold_bay_3857_clip.tiff"
```

```
grid_side=10
command <- paste("raster2pgsql -d  -M  ",r_nm, " -F -t ",grid_side,"x",grid_side," tmp|psql -h postgis
system(command)

query<-"CREATE OR REPLACE FUNCTION median (float[]) RETURNS float AS '
x<-arg1
x<-as.numeric(as.character(x))
x<-na.omit(x)
median(x)'
LANGUAGE 'plr' STRICT;
CREATE OR REPLACE FUNCTION q10 (float[]) RETURNS float AS '
x<-arg1
x<-as.numeric(as.character(x))
x<-na.omit(x)
quantile(x,0.1,na.rm=TRUE)'
LANGUAGE 'plr' STRICT;
CREATE OR REPLACE FUNCTION q90 (float[]) RETURNS float AS '
x<-arg1
x<-as.numeric(as.character(x))
x<-na.omit(x)
quantile(x,0.9,na.rm=TRUE)'
LANGUAGE 'plr' STRICT;"

odbcQuery(con,query)
```

```
## [1] 1
```

```
query<-"drop table if exists tmp2;
create table tmp2 as
select rid, st_envelope(rast) geom,
q10((st_dumpvalues(rast)).valarray) q10,
median((st_dumpvalues(rast)).valarray) median,
q90((st_dumpvalues(rast)).valarray) q90
from tmp"

odbcQuery(con,query)
```

```
## [1] 1
```

```
getquery <- function(query) {
    query <- paste("create view temp_view as ", query, sep = "")
    odbcQuery(con, query)
    dsn<-"PG:dbname='brant' host='postgis' port=5432 user= 'docker'"
    result <- readOGR(dsn, "temp_view")
    odbcQuery(con, "drop view temp_view")
    return(result)
}

grd<-getquery("select * from tmp2 where q10<0")
```

```
## OGR data source with driver: PostgreSQL
## Source: "PG:dbname='brant' host='postgis' port=5432 user= 'docker'", layer: "temp_view"
## with 3040 features
## It has 4 fields
```

```r
plot(grd)
```