Kandakji et al. - Identifying and Characterizing dust point sources in the southwestern United States using remote sensing and GIS

David Failing

2/9/2021

Links from Nick

- Work with MODIS in R: https://www.earthdatascience.org/courses/earth-analytics/multispectral-remote-sensing-modis/modis-data-in-R
- `MODIS` package in R: https://cran.r-project.org/web/packages/MODIS/MODIS.pdf

Links from the paper

- Tarek Kandakji OrcID: https://orcid.org/0000-0001-9173-5747
- Blowing dust paper: https://www.sciencedirect.com/science/article/pii/S0048969717328334?via%3Dihub
- Drought level and land use on dust: https://www.essoar.org/doi/10.1002/essoar.10500387.1
- 2021 paper: https://www.sciencedirect.com/science/article/pii/S0048969720359908
- The original paper: https://www.sciencedirect.com/science/article/pii/S0169555X19305100
- This file contains the location of the detected dust point sources: https://data.mendeley.com/datasets/6f8nyyr6n9/1

Figures

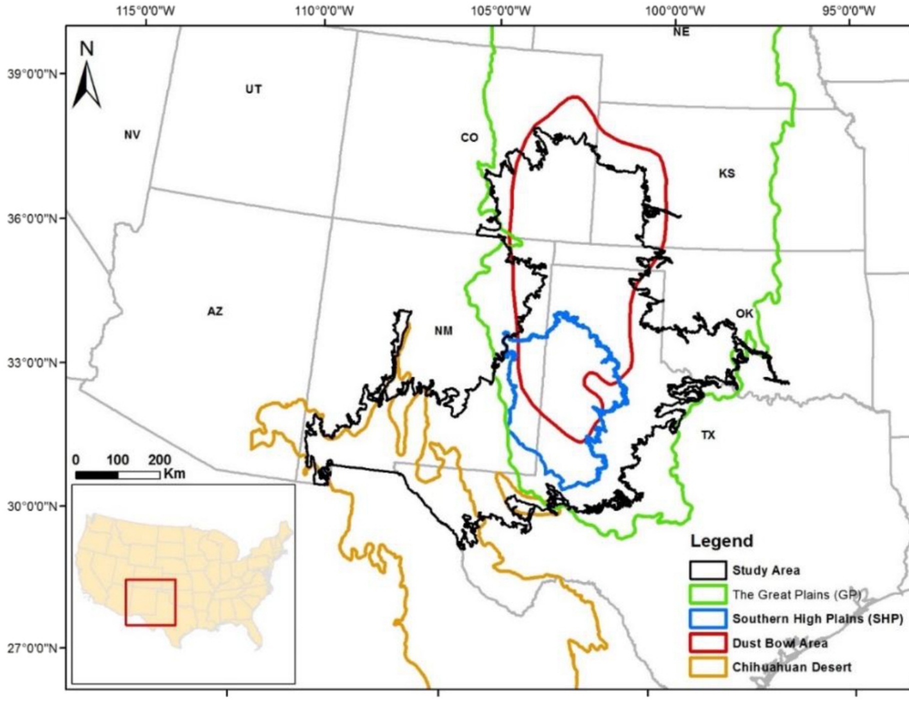

Figure 1: Study domain including key geographic regions discussed within the study.

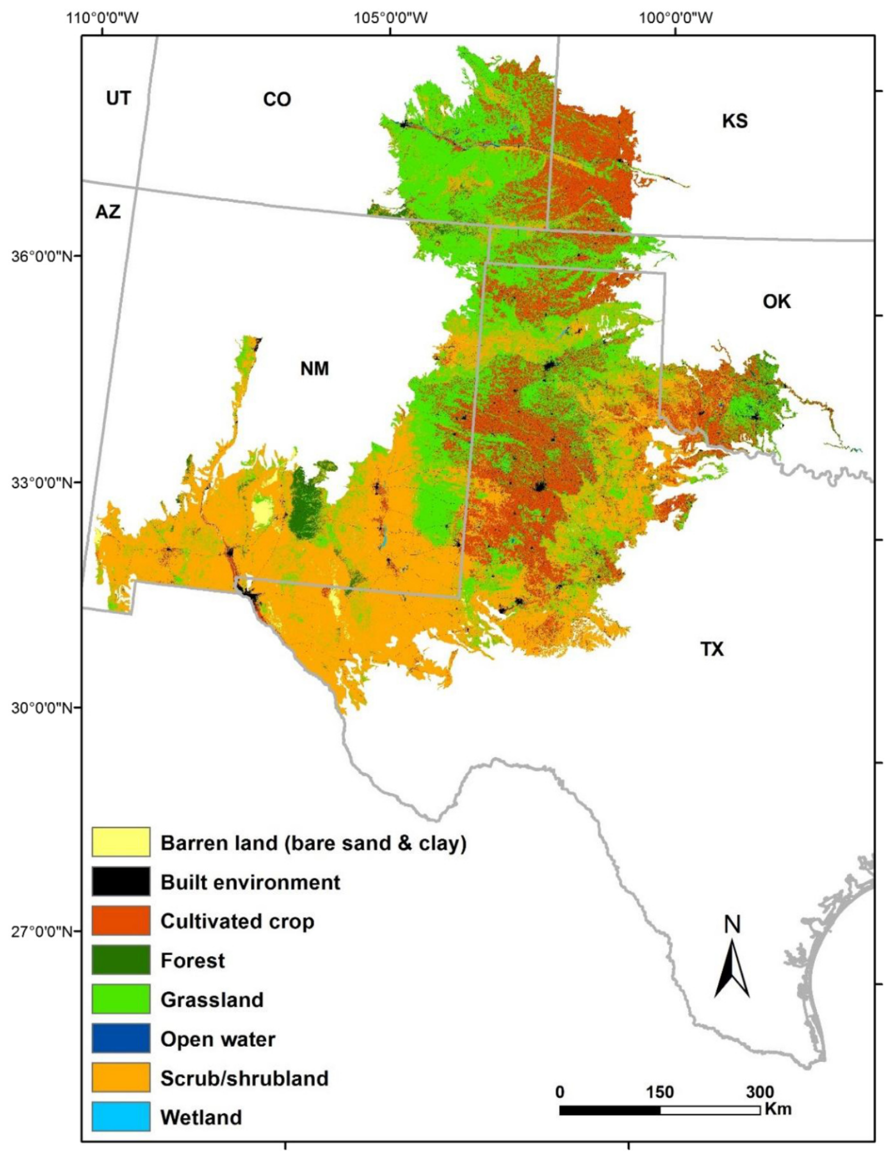

Figure 2: Land-cover map of the study domain for the year 2011. Data from Homer et al. (2015).

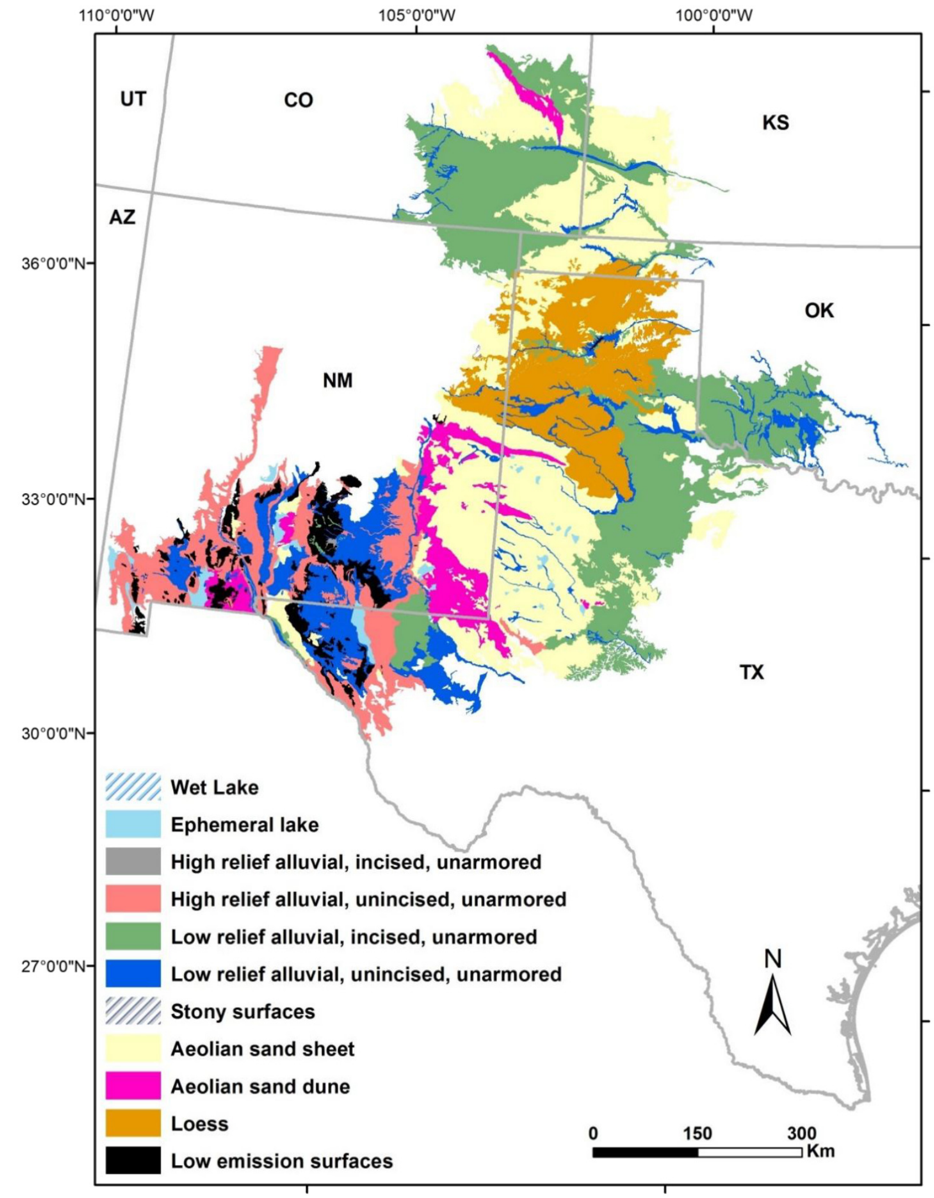

Figure 3: Geomorphology map of the domain, using the categories of Bullard et al. (2011) relating geomorphology to dust emission.

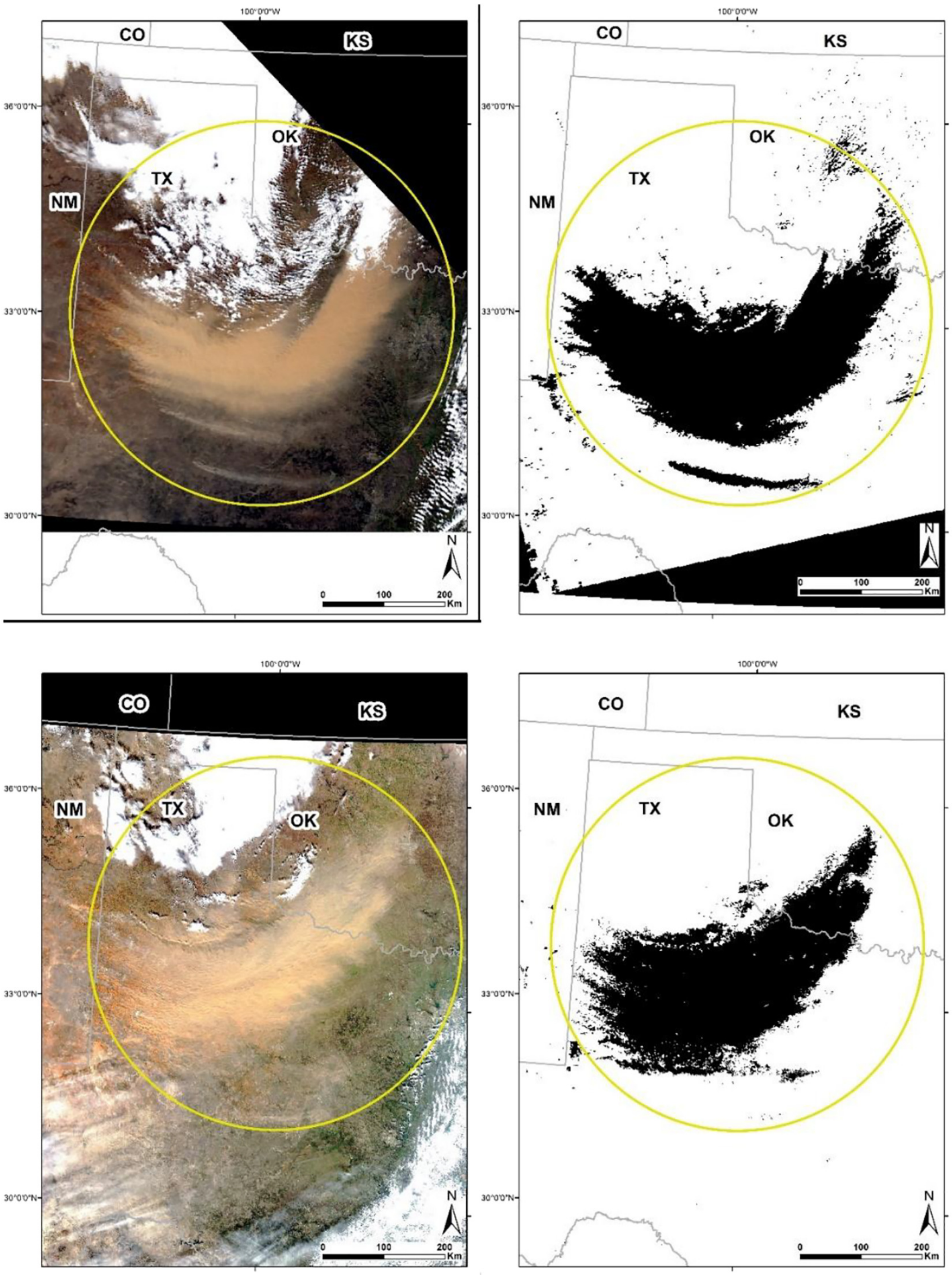

Figure 4: MODIS true color images (left) and their associated BTD images (right), for the days February 24, 2007 (top), and January 22, 2012 (bottom). The black color in the BTD images indicates dust.

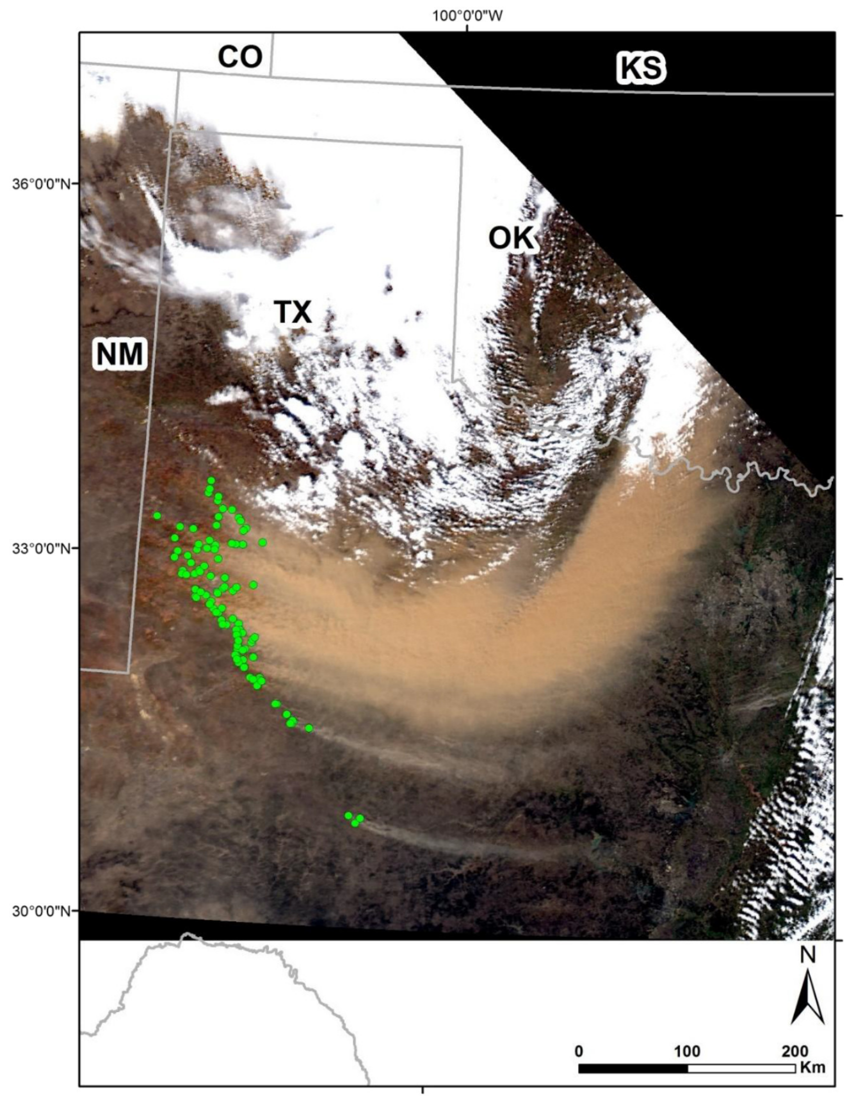

Figure 5: Dust source points identified for the dust event on February 24, 2007.

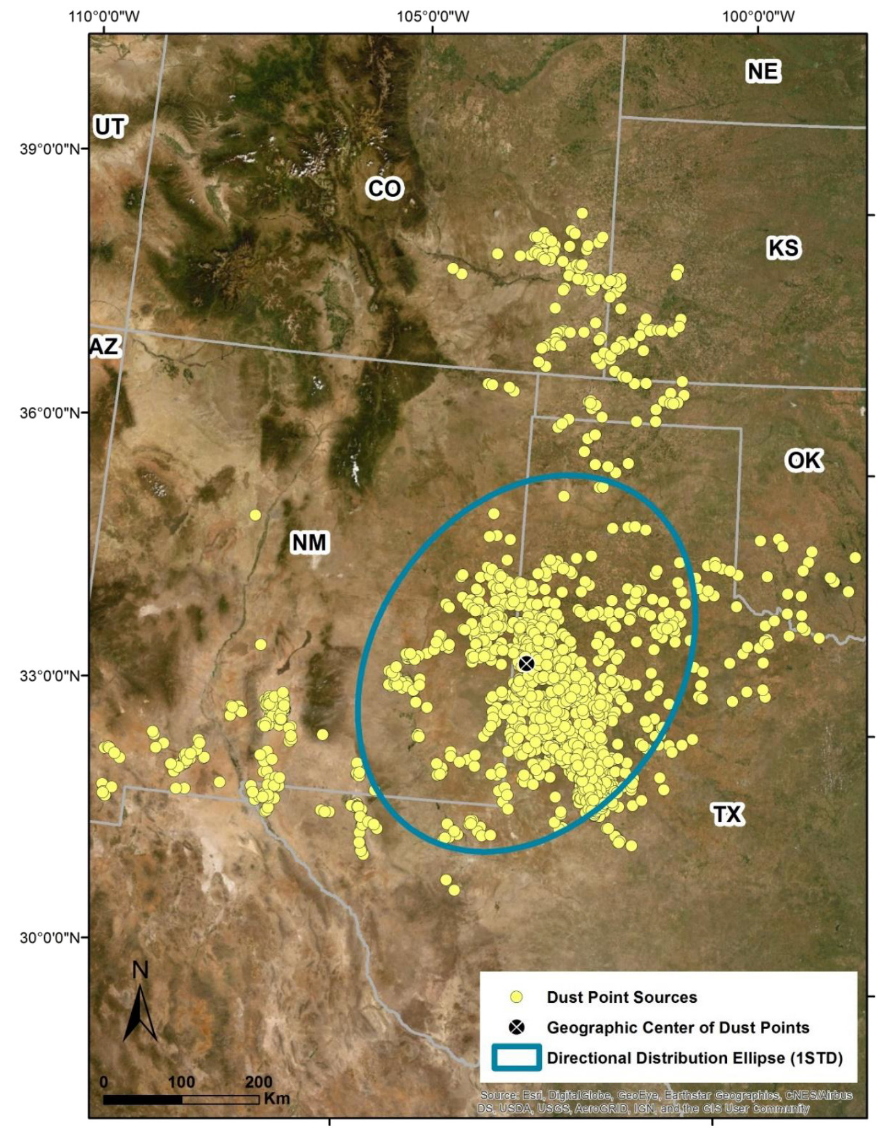

Figure 6: Mean center (geographic center) and directional distribution (Standard Deviation Ellipse) of all the dust points.

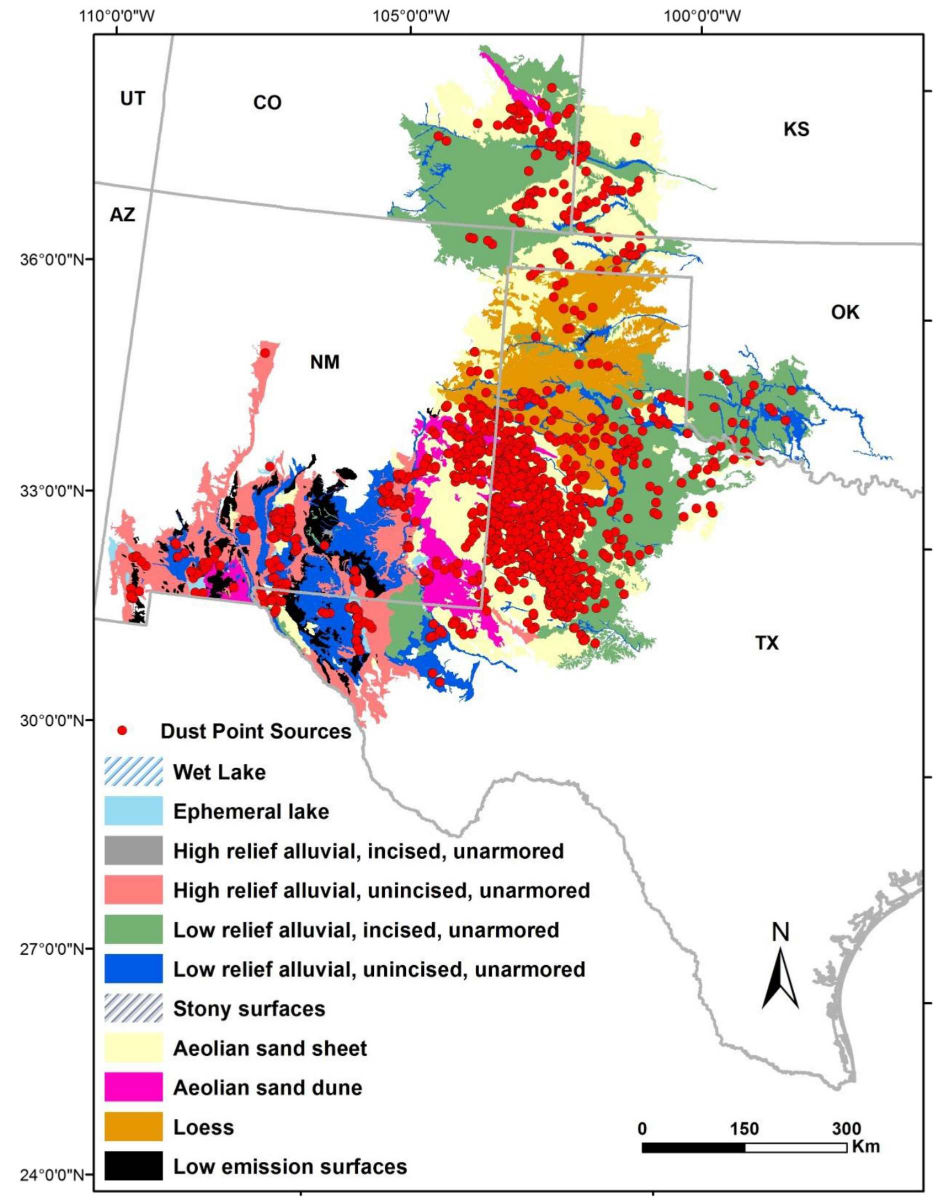

Figure 7: Identified dust source points (red dots) distributed over the geomorphic classes.

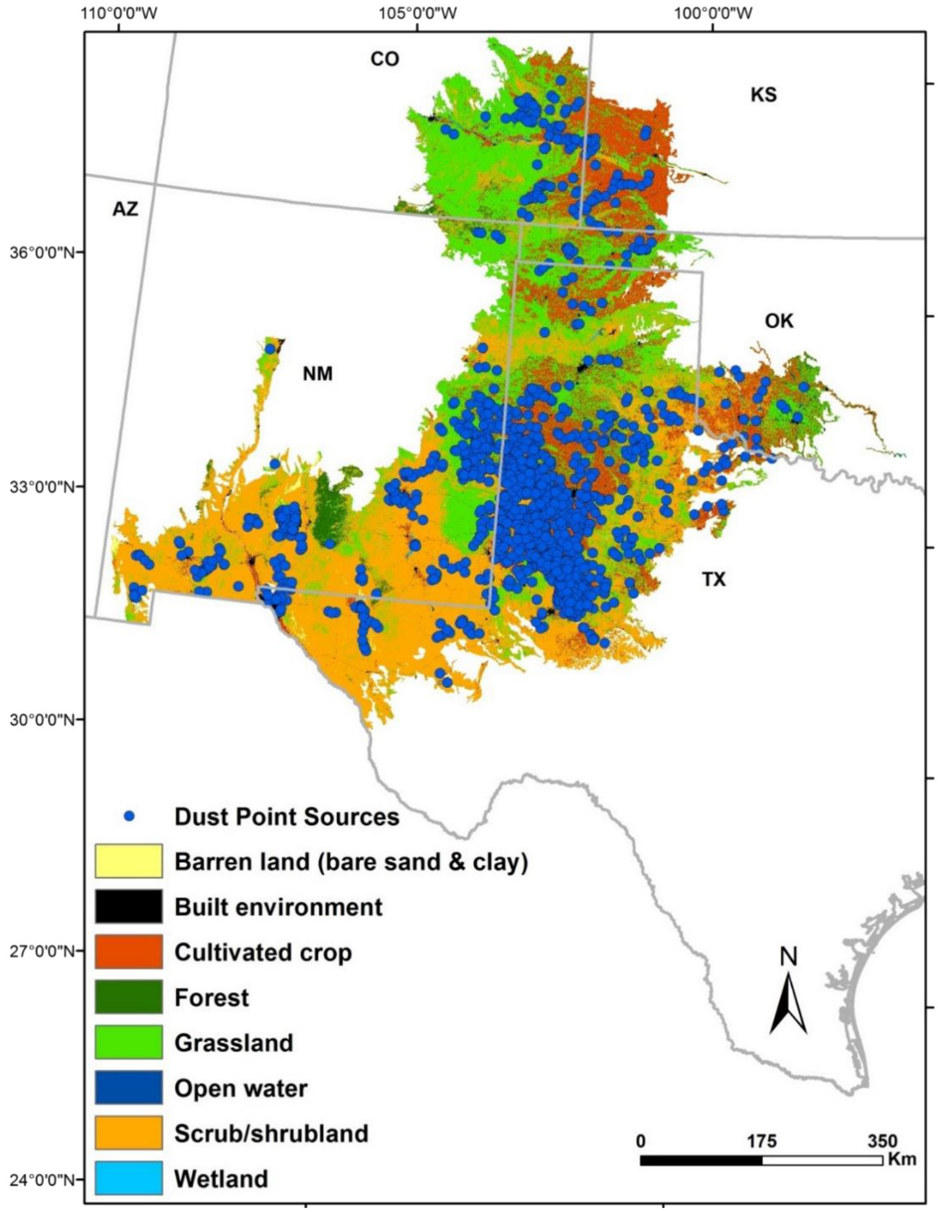

Figure 8: Identified dust source points distributed over the land-cover types of the study area.

Tables

Table 1. Weather stations used in the study.

```r
# Suppress code because this:
# The mutate contains str_replace(.$"Long., Lat.", "-",
# last of which is an unprintable character (Unicode minus)
table_1 <- readxl::read_excel("figures/Table-1.xlsx")

table_1 <- table_1 %>%
  rename(LongLat = 'Long., Lat.') %>%
  mutate(LongLat = str_replace(LongLat, "-", "-")) %>%
  extract(col = "Station name (station code)",
          into = c("Station_name", "Station_code"),
          regex = "(.+) \\((.+)\\)", remove = TRUE) %>%
  separate(col = LongLat, into = c("Longitude", "Latitude"),
         sep = ", ", remove = TRUE, convert = TRUE)

# Store the dates for the earliest available in 2001-16 at each site. (As of 2021-02-18)
first_avail <- c("2007-01-26", "2001-01-01", "2001-01-01", "2001-01-01", "2001-01-01",
                 "2001-01-01", "2005-06-01", "2001-01-02", "2014-11-06", "2001-01-01",
                 "2001-01-01", "2001-01-01", "2001-01-01", "2001-01-01", "2001-01-01")
table_1 <- cbind(table_1, first_avail)

rm(first_avail)

## Write final table to CSV for later loading
write.csv(table_1, file = "data/table_1_cleaned.csv",
          row.names = FALSE)
```

Table 2. Summary of MODIS products used in the study.

```
table_2 <- readxl::read_excel("figures/Table-2.xlsx")

table_2 <- table_2 %>%
  na_if(., "N/A") %>%
  separate(col = "MODIS product",
           into = c("MODIS_Terra_product", "MODIS_Aqua_product"),
           sep = " & ", remove = TRUE) %>%
  extract(col = "Bands", into = c("Band_number", "Band_name"),
          regex = "Band ([[:digit:]]+) \\((.+)\\)", remove = TRUE)

## Write final table to CSV for later loading
write.csv(table_2, file = "data/table_2_cleaned.csv",
          row.names = FALSE)
```

Table 3. Summary of dust emission analysis by geomorphic class.

```r
table_3 <- readxl::read_excel("figures/Table-3.xlsx")

table_3 <- table_3 %>%
  setNames(c("Geomorphology_class", "Area_km2", "Area_pct",
            "Dust_src_pts_n", "Dust_src_pts_pct",
            "Dust_Emission_Ratio_DER")) %>%
  na_if(., "-") %>%
  mutate(Subregion = ifelse(is.na(Geomorphology_class),
                            Geomorphology_class[-1],"Study Area")) %>%
  relocate(Subregion , .after = Geomorphology_class) %>%
  filter(!(Geomorphology_class %in%
             c("Great Plains", "Southern High Plains", "Chihuahuan Desert"))) %>%
  fill(Geomorphology_class, .direction = "down") %>%
  mutate(Area_pct = ifelse( Area_pct == "1.1000000000000001", "1.1", Area_pct)) %>%
  mutate(Area_pct = ifelse( Area_pct == "<0.1", "0", Area_pct)) %>%
  mutate(Area_pct = as.numeric(Area_pct)) %>%
  mutate(DER_comp = ifelse(Dust_src_pts_pct == 0, 0, Dust_src_pts_pct / Area_pct)) %>%
  mutate(DER_comp = ifelse(Subregion == "Study Area", NA, DER_comp))

## Write final table to CSV for later loading
write.csv(table_3, file = "data/table_3_cleaned.csv",
          row.names = FALSE)
```

Table 4. Summary of dust emission analysis by land cover type.

```
table_4 <- readxl::read_excel("figures/Table-4.xlsx")

table_4 <- table_4 %>%
  setNames(c("Land_cover_type", "Area_km2", "Area_pct",
             "Dust_src_pts_n", "Dust_src_pts_pct",
             "Dust_Emission_Ratio_DER")) %>%
  na_if(., "-") %>%
  mutate(Subregion = ifelse(is.na(Land_cover_type),
                            Land_cover_type[-1],"Study Area")) %>%
  relocate(Subregion , .after = Land_cover_type) %>%
  filter(!(Land_cover_type %in%
           c("Great Plains", "Southern High Plains", "Chihuahuan Desert"))) %>%
  fill(Land_cover_type, .direction = "down") %>%
  mutate(Area_pct = ifelse( Area_pct == "2.2000000000000002", "2.2", Area_pct)) %>%
  mutate(Area_pct = ifelse( Area_pct == "20.399999999999999", "20.4", Area_pct)) %>%
  mutate(Area_pct = ifelse( Area_pct == "<0.1", "0", Area_pct)) %>%
  mutate(Area_pct = as.numeric(Area_pct)) %>%
  mutate(DER_comp = ifelse(Dust_src_pts_pct == 0, 0, Dust_src_pts_pct / Area_pct)) %>%
  mutate(DER_comp = ifelse(Area_pct == 0, NA, Dust_src_pts_pct / Area_pct)) %>%
  mutate(DER_comp = ifelse(Subregion == "Study Area", NA, DER_comp))

## Write final table to CSV for later loading
write.csv(table_4, file = "data/table_4_cleaned.csv",
          row.names = FALSE)
```

Methodology

COMMENT: I think the order might have been 3.3 -> -> 3.4 -> 3.5 -> 3.1 -> 3.2, with the study area defined after they identified dust sources. The jagged borders of the black Study Area in Figure 1 seem to indicate this. Doing 3.3 first makes the most sense.

3.1 Land-cover map

"The data for the land-cover map (Fig. 2) were obtained from the National Land Cover Database (Homer et al., 2015) via their website: https://mrlc.gov. Maps for 2011 were used in this study."

3.2 Geomorphology

"A geomorphology map (Fig. 3) was created according to the dust source geomorphic classification adopted by Bullard et al. (2011). The classification was applied to a soil base map obtained from the USA Gridded Soil Survey Geographic (gSSURGO) Database State-tile Package (https://gdg.sc.egov.usda.gov/GDGOrder.aspx?order=QuickState)."

"The adjoining common polygons across each state's borders were unified. Based on the geomorphological information provided by the soil map, a geomorphic class from Bullard et al. (2011) was assigned to that polygon. The borders of the study area, however, were determined after the dust points were identified. The continuous outline of all polygons that included at least one dust point determined the border of the domain as shown in Fig. 1."

3.3 Weather data collection

"Weather data were collected from 15 meteorological stations located in five states: Colorado, Kansas, New Mexico, Oklahoma, and Texas, as were also used by Li et al. (2018) (Table 1). The data were obtained for the years 2011 to 2016 (https://mesonet.agron.iastate.edu/request/download.phtml)."

We can use the package `pmetar` in R to download and parse current METAR reports. Note that `pmetar::metar_get` is able to download multiple codes at once through an input vector, and codes are not case sensitive. We can also use the package `riem` to get data. `pmetar` was published 2021-01-13 so it is brand spanking new, while `riem` was published 2016-09-10.

```r
# Print the most recent weather for all station codes in Table 1.
# NOTE: pmetar appears to have changed to require only a lone, not a vector, input here
for (Station in table_1$Station_code) {
  print(pmetar::metar_get(Station))
}
```

```
## [1] "KLAA 251953Z AUTO 22007G16KT 10SM CLR 13/M01 A2968 RMK AO2 SLP045 T01331011"
## [1] "KDDC 251952Z AUTO 26008KT 10SM BKN022 OVC030 07/02 A2980 RMK AO2 SLP099 T00720022"
## [1] "KGCK 252012Z 19004KT 10SM BKN032 09/02 A2977 RMK AO2 T00890017"
## [1] "KCVN 251956Z AUTO 25022G31KT 10SM CLR 15/M05 A2979 RMK AO2 PK WND 26031/1949 SLP082 T01501050"
## [1] "KHOB 251950Z 24009G17KT 10SM SKC 16/M05 A2986"
## [1] "KLRU 252035Z AUTO 21012G22KT 10SM SCT065 17/M03 A2990 RMK AO2 T01691032"
## [1] "KAXS 252035Z AUTO 27009G17KT 10SM CLR 16/02 A2984 RMK AO2 T01570024"
## [1] "KLTS 251956Z AUTO 26009KT 10SM CLR 15/03 A2984 RMK AO2 SLP102 T01530025 $"
## [1] "KELK 252035Z AUTO 27008KT 250V310 10SM CLR 13/02 A2985 RMK AO2 T01280015"
## [1] "KCSM 251953Z 27010KT 10SM CLR 12/04 A2983 RMK AO2 SLP098 T01220039"
## [1] "KGUY 251953Z AUTO 16007KT 10SM CLR 11/01 A2977 RMK AO2 SLP073 T01110011"
## [1] "KAMA 251953Z 24015G23KT 10SM CLR 14/M01 A2976 RMK AO2 SLP065 T01441006"
## [1] "KELP 251951Z 24013G20KT 10SM FEW085 FEW250 16/M05 A2989 RMK AO2 SLP086 T01611050"
## [1] "KLBB 251953Z 25021G25KT 10SM FEW065 SCT250 17/M01 A2981 RMK AO2 SLP075 T01721011"
## [1] "KMAF 251953Z 28014G22KT 10SM CLR 18/M03 A2986 RMK AO2 PK WND 26026/1919 SLP088 T01781028"
```

```r
# Get the location information for stations in Table 1
# NOTE: Includes Elevation (m), where `table_1` excludes it.
pmetar::metar_location(table_1$Station_code)
```

```
## # A tibble: 15 x 7
##    ICAO_Code IATA_Code Airport_Name      Longitude Latitude Elevation Source
##    <chr>     <chr>     <chr>                 <dbl>    <dbl>     <dbl> <chr>
##  1 KLAA      LAA       Lamar Municipal ~    -103.      38.1     1130. http://ou~
##  2 KDDC      DDC       Dodge City Regio~    -100.      37.8      791. http://ou~
##  3 KGCK      GCK       Garden City Regi~    -101.      37.9      881. http://ou~
##  4 KCVN      CVN       Clovis Municipal~    -103.      34.4     1285. http://ou~
##  5 KHOB      HOB       Lea County Regio~    -103.      32.7     1116. http://ou~
##  6 KLRU      LRU       Las Cruces Inter~    -107.      32.3     1358. http://ou~
##  7 KAXS      AXS       Altus Quartz Mou~     -99.3     34.7      437. http://ou~
##  8 KLTS      LTS       Altus Air Force ~     -99.3     34.7      421. http://ou~
##  9 KELK      ELK       Elk City Regiona~     -99.4     35.4      614. http://ou~
## 10 KCSM      CSM       Clinton Sherman ~     -99.2     35.3      586. http://ou~
## 11 KGUY      GUY       Guymon Municipal~    -102.      36.7      952. http://ou~
## 12 KAMA      AMA       Rick Husband Ama~    -102.      35.2     1099. http://ou~
## 13 KELP      ELP       El Paso Internat~    -106.      31.8     1207. http://ou~
## 14 KLBB      LBB       Lubbock Preston ~    -102.      33.7     1000. http://ou~
## 15 KMAF      MAF       Midland Internat~    -102.      31.9      875. http://ou~
```

```r
# LAA says start is 1994-12-31 18:00:00-06 per
#     https://mesonet.agron.iastate.edu/sites/networks.php?network=CO_ASOS&format=html
# The below are Oklahoma stations
#     https://mesonet.agron.iastate.edu/sites/networks.php?network=OK_ASOS&format=html
# LTS in the same tool says 2001-01-01 missing but ok after
# AXS has a first date 2005-06-01
# ELK has a first date 2014-11-06 08:15:00-06

for(i in table_1$Station_code[-c(1,7:9)]){
   assign(paste0("METAR_", i),
          do.call(pmetar::metar_get_historical,
                  list(i, start_date = "2001-01-01",
                       end_date = "2001-01-31",
                       from = "iastate")) )
}

# Lubbock is# Lubbock is in West Texas, where many dust sources are, and can be
METAR_LBB <- pmetar::metar_get_historical("LBB",
                              start_date = "2001-01-01",
                              end_date = "2016-12-31",
                              from = "iastate")

metar_dust_abbrev <- c("DS", "DU", "BLDU", "DRSA", "DS", "PO", "SA", "SS",
                       "VCBLDU", "VCDS", "VCPO", "VCSS")

sum(grepl(c("BLDU| DU"), METAR_LBB)) / length(METAR_LBB)
```

```
## [1] 0.01260614
```

```r
# This is a BLDU event that would get missed by `pmetar` because it ignores the remarks.
"200103151753 METAR METAR KLBB 151753Z 33030G39KT 9SM CLR 12/M06 A2993 RMK AO2 PK WND 33041/1741 SLP117
```

```
## [1] "200103151753 METAR METAR KLBB 151753Z 33030G39KT 9SM CLR 12/M06 A2993 RMK AO2 PK WND 33041/1741
```

```r
metar_wx_codes("200103151753 METAR METAR KLBB 151753Z 33030G39KT 9SM CLR 12/M06 A2993 RMK AO2 PK WND 33
```

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

```r
#Let's try to use `riem` to investigate the same date range and see what their functions do.
# NOTE: date_end should be the day AFTER the last day on which you want observations
# DOCUMENTATION IS AT https://docs.ropensci.org/riem/reference/riem_measures.html

METAR_KLBB <- riem::riem_measures(station = "KLBB", date_start = "2001-01-01", date_end = "2017-01-01")
METAR_KLBB_03_15 <- METAR_KLBB %>%
  filter(grepl("2001-03-15", as.character(valid)))

# View(METAR_KLBB_03_15[18:22,c("wxcodes", "metar")])# Rows showing discrepancy

METAR_KLBB %>% filter(grepl("2016-12-31", as.character(valid)))
```

```
## # A tibble: 313 x 31
##    station valid                  lon   lat  tmpf  dwpf  relh  drct  sknt  p01i
##    <chr>   <dttm>               <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
##  1 LBB     2016-12-31 00:00:00 -102.   33.7    48    32  53.7   210    12     0
##  2 LBB     2016-12-31 00:05:00 -102.   33.7    NA    NA  NA     210    14    NA
##  3 LBB     2016-12-31 00:10:00 -102.   33.7    NA    NA  NA     210    12    NA
```

```
##  4 LBB     2016-12-31 00:15:00 -102.  33.7    NA    NA  NA     200    12    NA
##  5 LBB     2016-12-31 00:20:00 -102.  33.7    NA    NA  NA     200    12    NA
##  6 LBB     2016-12-31 00:25:00 -102.  33.7    NA    NA  NA     200    12    NA
##  7 LBB     2016-12-31 00:30:00 -102.  33.7    NA    NA  NA     200    13    NA
##  8 LBB     2016-12-31 00:35:00 -102.  33.7    NA    NA  NA     200    13    NA
##  9 LBB     2016-12-31 00:40:00 -102.  33.7    NA    NA  NA     200    14    NA
## 10 LBB     2016-12-31 00:45:00 -102.  33.7    NA    NA  NA     200    16    NA
## # ... with 303 more rows, and 21 more variables: alti <dbl>, mslp <dbl>,
## #   vsby <dbl>, gust <dbl>, skyc1 <chr>, skyc2 <chr>, skyc3 <chr>, skyc4 <chr>,
## #   skyl1 <dbl>, skyl2 <dbl>, skyl3 <dbl>, skyl4 <dbl>, wxcodes <chr>,
## #   ice_accretion_1hr <dbl>, ice_accretion_3hr <dbl>, ice_accretion_6hr <dbl>,
## #   peak_wind_gust <dbl>, peak_wind_drct <dbl>, peak_wind_time <chr>,
## #   feel <dbl>, metar <chr>
```

```
METAR_KLBB[,c("valid", "vsby", "wxcodes", "metar")] %>% filter(grepl("BLDU", wxcodes))
```

```
## # A tibble: 1,440 x 4
##    valid                vsby wxcodes metar
##    <dttm>              <dbl> <chr>   <chr>
##  1 2001-01-13 22:53:00   3   BLDU    METAR KLBB 132253Z 27028G35KT 3SM BLDU CLR~
##  2 2001-01-13 23:53:00   3   BLDU    METAR KLBB 132353Z 28021G29KT 3SM BLDU CLR~
##  3 2001-02-24 17:53:00   3   BLDU    METAR KLBB 241753Z 26033G43KT 3SM BLDU CLR~
##  4 2001-02-24 18:53:00   1   BLDU    METAR KLBB 241853Z 25037G45KT 1SM BLDU CLR~
##  5 2001-02-24 19:53:00   0.5 BLDU    METAR KLBB 241953Z 24032G45KT 1/2SM BLDU V~
##  6 2001-02-24 20:53:00   0.5 BLDU    METAR KLBB 242053Z 26035G47KT 1/2SM BLDU V~
##  7 2001-02-24 21:53:00   0.5 BLDU    METAR KLBB 242153Z 26033G45KT 1/2SM BLDU V~
##  8 2001-02-24 22:53:00   0.5 BLDU    METAR KLBB 242253Z 27031G38KT 1/2SM BLDU V~
##  9 2001-02-24 23:53:00   0.5 BLDU    METAR KLBB 242353Z 26026G35KT 1/2SM BLDU V~
## 10 2001-02-25 00:53:00   0.5 BLDU    METAR KLBB 250053Z 26020KT 1/2SM BLDU VV01~
## # ... with 1,430 more rows
```

3.3 DATA PULL

```
# Download METAR data using package `riem`
# Export METAR data to .csv
# TO DO LATER: Delete downloaded METAR data from R Environment. We'll read it in one at a time later.
# NOTE: date_end should be the day AFTER the last day on which you want observations
# DOCUMENTATION IS AT https://docs.ropensci.org/riem/reference/riem_measures.html

for(i in 1:nrow(table_1)){
  filename <- paste0("METAR_", table_1[i, "Station_code"])
  assign(filename,
         do.call(riem::riem_measures,
                 list(station = table_1[i, "Station_code"],
                      date_start = table_1[i, "first_avail"],
                      date_end = "2017-01-01")) )
  do.call(write.csv, list(x = as.name(filename),
                          file = paste0("data/", filename,".csv"),
                          row.names = FALSE))
rm(list = filename)
}

rm(filename, i)
```

```r
# OLD
# Download METAR data from API.
# for(i in 1:2){
#    assign(paste0("METAR_", table_1[i, "Station_code"], "_PMETAR"),
#           do.call(pmetar::metar_get_historical,
#                   list(table_1[i, "Station_code"], start_date = table_1[i, "first_avail"],
#                        end_date = "2016-12-31",
#                        from = "iastate")) )
# }
```

"Dusty days were determined following the procedure by Lee et al. (2012). A dust event was recorded if the visibility in one or more stations was less than 3 mi (~5km) for at least 1 h and the present weather code was indicating dust, or the weather code was dust for a minimum of 2h regardless of visibility."

"The METAR present weather codes (https://www.weather.gov/media/wrh/mesowest/metar__decode__key.pdf) that were considered as dust are: DS, DU, BLDU, DRSA, DS, PO, SA, SS, VCBLDU, VCDS, VCPO, and VCSS. Using the 3 mi visibility criterion or the 2 h dust reporting increase the probability that blowing dust is dense enough to be detected by the satellite imagery (Mahowald et al., 2003; She et al., 2018)"

Note that a subset of these codes are already stored (with additional information) in `pmetar::metarWXcodes`.

Alright, some of the explanation of what METAR data looks like is at https://en.wikipedia.org/wiki/METAR (METAR stands for Meteorological Aerodrome Report) too, as well as in the KEY TO DECODE AN ASOS OBSERVATION at the end of the following PDF.

```r
# METAR/TAF LIST OF ABBREVIATIONS AND ACRONYMS - Reading the PDF into R
# Only the first 3 pages contain our abbreviations.
# The last 2 pages contain a "KEY TO DECODE AN ASOS (METAR) OBSERVATION"
# which might be of use later on.

# Use tabulizer::locate_areas() to get graphical location of desired columns.
# areas are specified a2 <- list(c(126, 149, 212, 462), c(126, 284, 174, 417))
# in this case for multiple pages.
# You can replace filename w/link
# https://www.weather.gov/media/wrh/mesowest/metar_decode_key.pdf

# You can use locate_areas to determine corners. Output is a list of vectors.
# out <- tabulizer::locate_areas("figures/z_metar_decode_key.pdf", c(1, 1, 2, 2, 3, 3))

## Save the corner points in a list
corners <- list(c(top = 0, left = 0, bottom = 1000, right = 1000), #p1
                c(top = 50, left = 390, bottom = 761, right = 770), #p1
                c(top = 16, left = 6, bottom = 776, right = 320), #p2
                c(top = 16, left = 390, bottom  = 776, right = 770), #p2
                c(top = 16, left = 6, bottom = 486, right = 390), #p3
                c(top = 16, left = 390, bottom = 486, right = 770)) #p3

# Use tabulizer::extract_tables to pull tables from the PDF
out <-tabulizer::extract_tables(file = "data/z_metar_decode_key.pdf",
                                pages = c(1,1,2,2,3,3),
                                area = corners[1:6],
                                method = "lattice")
## Page 1
out[[1]] <- out[[1]][-1,]
out[[1]][1, 1:2] <- out[[1]][1, 2:3]
```

```r
sec_1 <- as.data.frame(out[[1]][-22,1:2]) %>%
  setNames(c("Abbrev", "Def"))

sec_2 <- as.data.frame(out[[2]][-28,]) %>%
  setNames(c("Abbrev", "Def"))

## Page 2
out[[3]][1,1:2] <- out[[3]][1,3:4]

sec_3 <- as.data.frame(out[[3]][,1:2]) %>%
  setNames(c("Abbrev", "Def"))

sec_4 <- as.data.frame(out[[4]]) %>%
  setNames(c("Abbrev", "Def"))

## Page 3
sec_5 <- as.data.frame(out[[5]][,1:2]) %>%
  setNames(c("Abbrev", "Def"))

sec_6 <- as.data.frame(out[[6]]) %>%
  setNames(c("Abbrev", "Def"))

## Combining sections into a single data frame
metar_abbrev <- rbind(as.name(paste0("sec_", 1:6)))

metar_abbrev <- do.call(rbind, lapply( paste0("sec_", 1:6) , get)) %>%
  arrange(Abbrev)

## Write final table to CSV for later loading
write.csv(metar_abbrev, file = "data/METAR-Abbreviations.csv",
          row.names = FALSE)

## Clean up input files, leaving data.frames only
rm(list = c("corners", "out", grep("sec_", ls(), value = TRUE)))

# Note authors say 4 char abbrev that are 2x2 char in table
# "BLDU" but this is "BL" and "DU",
# "DRSA" is "DR" and "SA", and "VC" is a prefix for "in the vicinity"

metar_dust_abbrev <- c("DS", "DU", "BLDU", "DRSA", "DS", "PO", "SA", "SS",
                       "VCBLDU", "VCDS", "VCPO", "VCSS")

metar_dust_df <- metar_abbrev %>%
  filter(Abbrev %in% c("DS", "DU", "BL", "DU", "DR", "SA", "DS", "PO", "SA", "SS",
                       "VC", "BL", "DU", "VC", "DS", "VC", "PO", "VC", "SS") )
```

3.3 DATA FILTERING

```r
# A list of acceptable dust abbreviations from the paper
metar_dust_abbrev <- sort(c("DS", "DU", "BLDU", "DRSA", "DS", "PO", "SA", "SS",
                       "VCBLDU", "VCDS", "VCPO", "VCSS"))
metar_dust_regex <- paste(metar_dust_abbrev, collapse = "|")
```

```r
# List of files for further analysis
METAR_files <- list.files(path = "data/",
                          pattern = "METAR_.{3}\\.csv",
                          full.names = TRUE)

# Filter the data on entries with dust codes.
# GREP isn't great for this - TSSN is one wxcode that shows up w/o dust.
for(file in METAR_files){
  METAR_data <- read.csv(file) %>%
    subset(grepl(metar_dust_regex, wxcodes))
  write.csv(METAR_data, gsub("\\.csv", "_filtered\\.csv", file))
}
rm(METAR_data, file) # Clean up after looping

# Clean up at the end of the block
rm(metar_dust_abbrev, metar_dust_regex, METAR_files)
```

3.4 MODIS data collection

"MODIS true color images were obtained from the University of Wisconsin Space Science and Engineering Center (SSEC) (http://ge.ssec.wisc.edu/modis-today/) and were examined visually prior to any analysis to detect visibility of dust plumes. Corrected MODIS true color images with 250 m spatial resolution were used. The website, however, has data available for dates after October 23, 2007. Moreover, some days after that date are missing from the website and not available. Subsequently, MODIS true color image for the missing dates were examined through the Level-1 and Atmosphere Archive & Distribution System (LAADS) Distributed Active Archive Center (DAAC) website (https://ladsweb.modaps.eosdis.nasa.gov/search/). For the missing days, true color images were created using the visible bands included in MODIS products MOD09GA and MYDO9GA, also obtained from LAADS."

"For each dusty day reported, the MODIS true color image for that day was examined to determine its suitability for analysis. Accordingly, if the image contained clouds over the dust source region and/or the dust plumes were identifiable, then that day was considered not suitable for image processing and thus excluded from the analysis (Lee et al., 2012)."

"After determining which dusty days had MODIS images suitable for analysis, certain MODIS products for that day were downloaded from the LAADS website. The products included are summarized in Table 2."

3.5 Image Processing

3.5.1 Producing MODIS true color images

3.5.2 Processing MODIS thermal bands

3.6 Identifying dust point sources

3.7 Calculating Dust Emission Ratio

3.8 Cluster analysis