Distribution of structures over time

Author: Alistair Grom 964398

Getting started

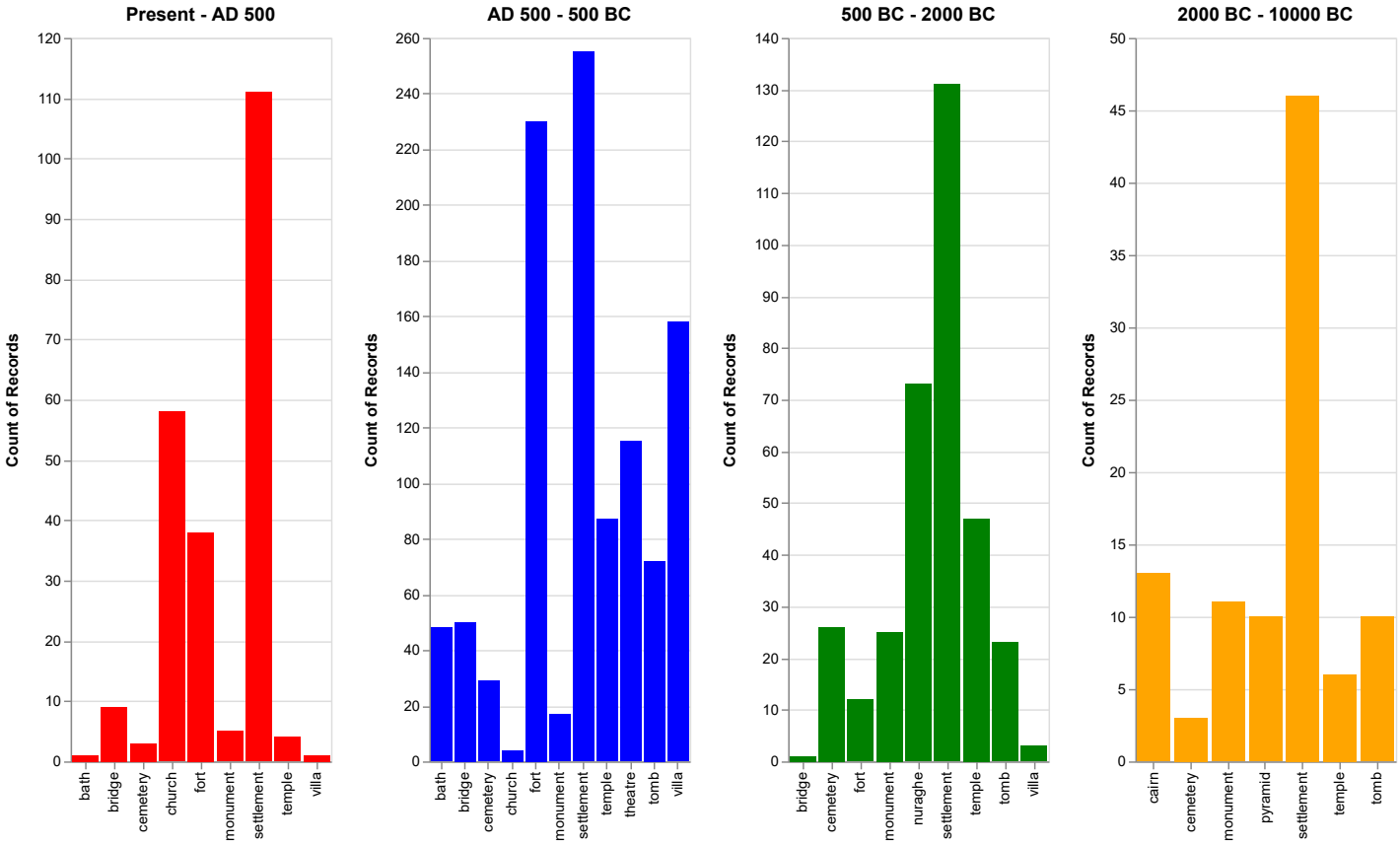
What can we learn from the visualization?

The Aim of this visualisation is to show how the distribution of structures has changed over the time periods from present to -10000 BC (or the start of the dataset). This can provide us with insights into what period we are in (roman, greek, egypytian).

From this visualisation we can see how the frequency of settlements has grown up until 500 AD.
These graphs also show how the respective empires constructed varous structures. The egyptians showing pyramids constructed until 2000 BC, we see that technology has improved in the 500 BC - 2000 BC period, with the introduction of bridges and nuraghes. A general increase in the amount of settlements also.
From 500 BC to AD 500 is where we see a great rise in productivity, likely due to the roman and greek empires being active during these times. A increase in forts, bridges, theatres, temples and so on.

From 500 AD the dataset indicates the main focus of society was to create settlements and churches with less emphasis on innovation.

```
barChart1 : Spec
barChart1 =
    let
        h =
            500

        w =
            180

        trans =
            transform
                --could be better to find the top 10 types for each filter
                << filter (fiExpr """datum.featureType == 'fort' ||
                                     datum.featureType == 'villa' ||
                                     datum.featureType == 'tomb' ||
                                     datum.featureType == 'cemetery' ||
                                     datum.featureType == 'pyramid' ||
                                     datum.featureType == 'cairn' ||
                                     datum.featureType == 'theatre' ||
                                     datum.featureType == 'church' ||
                                     datum.featureType == 'nuraghe' ||
                                     datum.featureType == 'temple'||
                                     datum.featureType == 'bridge' ||
                                     datum.featureType == 'bath' ||
                                     datum.featureType == 'monument' ||
                                     datum.featureType == 'settlement'
                                     """)

        trans1 =
            transform
                << filter (fiExpr """ (datum.maxDate <= 2100 && datum.minDate > 500) """)

        trans2 =
            transform
                << filter (fiExpr """ (datum.maxDate <= 500 && datum.minDate > -500) """)

        trans3 =
            transform
                << filter (fiExpr """ (datum.maxDate <= -500 && datum.minDate >= -2000) """)

        trans4 =
            transform
                << filter (fiExpr """ (datum.maxDate < -2000 && datum.minDate >= -10000) """)

        enc1 =
            encoding
                << position X [ pName "featureType", pTitle "" ]
                << position Y [ pAggregate opCount, pQuant ]

        spec1 =
            asSpec [ title "Present - AD 500" [], width w, height h, trans1 [], enc1 [], bar [ maColor "red", maOpacity 1 ] ]

        spec2 =
            asSpec [ title "AD 500 - 500 BC" [], width w, height h, trans2 [], enc1 [], bar [ maColor "blue", maOpacity 1 ] ]

        spec3 =
            asSpec [ title "500 BC - 2000 BC" [], width w, height h, trans3 [], enc1 [], bar [ maColor "green", maOpacity 1 ] ]

        spec4 =
            asSpec [ title "2000 BC - 10000 BC" [], width w, height h, trans4 [], enc1 [], bar [ maColor "orange", maOpacity 1 ] ]
    in
    toVegaLite
        [ locs
        , trans []
        , hConcat [ spec1, spec2, spec3, spec4 ]
        ]
```



What is the name for the type of visualization(s) used?

Bar chart

What are all visual mappings used?

Each bar chart has a w of 180 and h of 500.

y position

featureType category

length

count of location

Was there any special data preparation done?

For the records used here, replaced names that have trailing commas ('bath,' to 'bath' and so on)
Filtered out for the feature types I wanted
aggregated to get the count of these variables

What are the limitations of your design?

The scale is not the same on every graph, it would increase the sense of growth of structures and trends but the dataset doesnt contain the same amount of records for each time period.
This may be better as a toggle on or off but I was unable to do that.

Language Distribution from BC to AD

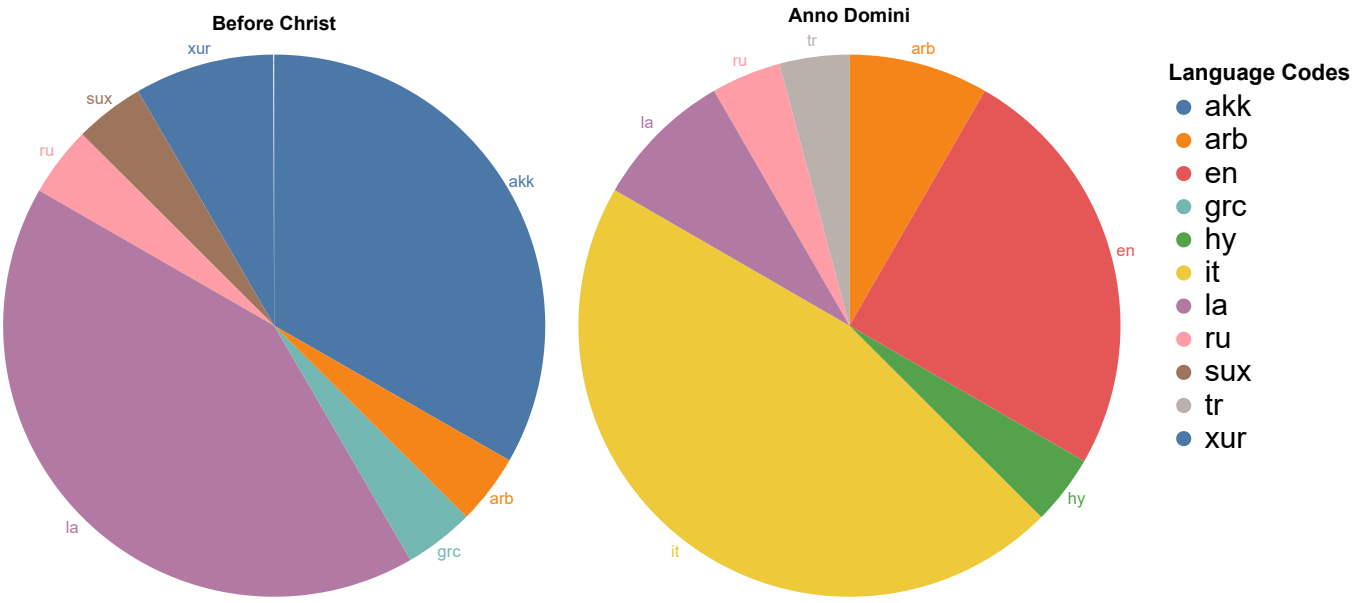
What can we learn from the visualization?

This visualisation of the language the name of a place/dataset entry is written in. This data allows us to understand the influence the particular culture had on the world in general in the given time period. If a place was particularly influential they would have a large proportion of the names in the dataset in their respective language.

From the data we have here we can see the trends for which languages have more ancient origins than others. Latin and Akkadian being 2 large languages before year 0. After year 0 we have italian as the largest followed not very closely by english.

An interesting occurence is that while you can see the Latin language areas move to Italian the same cannot be said for Akkadian, this could be due to it being one of the oldest languages in the Indo-Eurpoean region alongside Sumerian.

```
pieChart : Spec
pieChart =
    let
        cfg =
            configure
                << configuration (coView [ vicoStroke Nothing ])

        transBC =
            transform
                << filter (fiExpr "datum.nameLanguage != ''")
                << window [ ( [ wiAggregateOp opCount, wiField "nameLanguage" ], "langCountBC" ) ] []
                << filter (fiExpr "datum.langCountBC > 8950 && datum.minDate <= 0 ")

        transAD =
            transform
                << filter (fiExpr "datum.nameLanguage != ''")
                << window [ ( [ wiAggregateOp opCount, wiField "nameLanguage" ], "langCountAD" ) ] []
                << filter (fiExpr "datum.langCountAD > 8950 && datum.minDate > 0 ")

        encBC =
            encoding
                << position Theta
                    [ pName "langCountBC"
                    , pAggregate opSum
                    , pStack stZero
                    ]
                << color
                    [ mName "nameLanguage"
                    , mLegend
                        [ leTitle "Language Codes"
                        , leLabelFontSize 20
                        , leTitleFontSize 15
                        , leColumns 1
                        ]
                    ]

        encAD =
            encoding
                << position Theta
                    [ pName "langCountAD"
                    , pAggregate opSum
                    , pStack stZero
                    ]
                << color
                    [ mName "nameLanguage"
                    , mLegend
                        [ leTitle "Language Codes"
                        , leLabelFontSize 20
                        , leTitleFontSize 15
                        , leColumns 1
                        ]
                    ]

        pieSpec =
            asSpec [ arc [ maOuterRadius 180 ] ]

        labelEnc =
            encoding
                << text [ tName "nameLanguage" ]

        labelSpec =
            asSpec [ labelEnc [], textMark [ maRadius 190 ] ]

        viewBC =
            asSpec [ title "Before Christ" [], height 350, width 50, cfg [], names, transBC [], encBC [], layer [ pieSpec, lab

        viewAD =
            asSpec [ title "Anno Domini" [], height 350, width 50, cfg [], names, transAD [], encAD [], layer [ pieSpec, label
    in
    toVegaLite [ hConcat [ viewBC, viewAD ] ]
```



code	Language
akk	Akkadian
arb	Arabic
en	English
grc	Greek, Ancient (to 1453)
hy	Armenian
it	Italian
la	Latin
ru	Russian
sux	Sumerian
tr	Turkish
xur	Urartian

What is the name for the type of visualization(s) used?

Pie Chart

What are all visual mappings used?

theta nameLanguage count

Was there any special data preparation done?

The results with blank nameLanguages are filtered out

The names are aggregated and counted to form the field langCount which then is filtered with only languages with over (8950, number to get roughly 10 results) and by year > 0 and year < 0

What are the limitations of your design?

The largest language in this dataset is ' ' empty so this has an impact on the quality of the data.

Religious Sites location distribution across the World

What can we learn from the visualization?

This visualisation looks at buildings that have religous significance and their locations across the map. The purpose of this was to determine patterns in religion across the world.

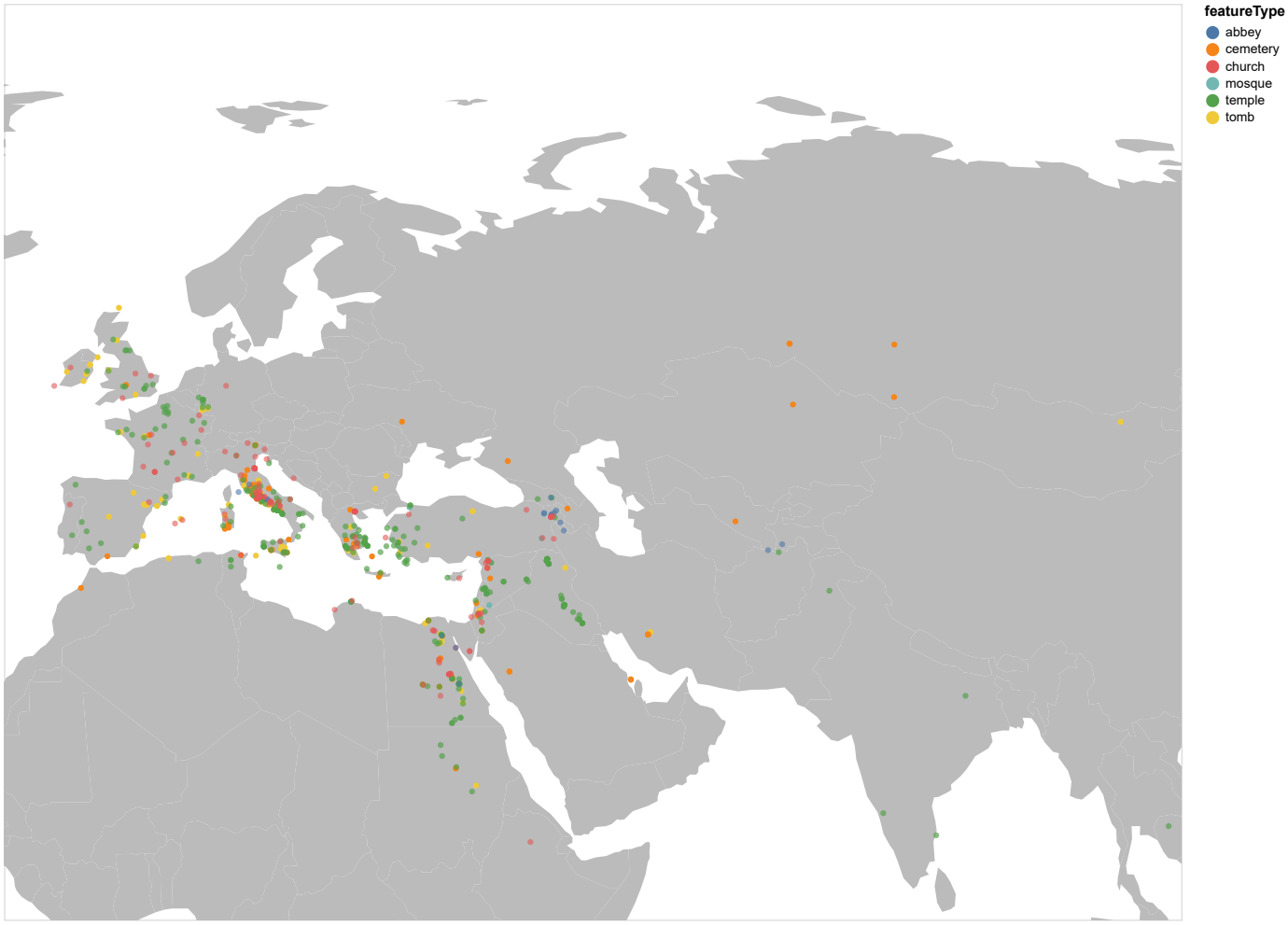
From this visualisation we learn that the dominant site is a temple, this is probably due to the Egyptians, Greeks and Romans all constructing buildings of this type for religious reasons.

The two zoomed in visualisations show both Greece and Italy/Rome being the most populus areas for these type of structure both with high concentrations of temples and Rome with a large amount of cemeterys.

This also allows us to see there are a small cluster of Abbeys in the Caucuses, which was a surprise to me. (possibly mislabelled in the dataset)

```
settlementEnc =
    encoding
        << position Latitude [ pName "reprLat" ]
        << position Longitude [ pName "reprLong" ]
        << color [ mName "featureType" ]


churchTrans =
    transform
        << filter (fiExpr """datum.featureType == 'church' ||
                             datum.featureType == 'abbey'
                             """)


templeTrans =
    transform
        << filter (fiExpr """datum.featureType == 'temple' ||
                            datum.featureType == 'temple,' """)


mosqueTrans =
    transform
        << filter (fiExpr """datum.featureType == 'mosque' ||
                            datum.featureType == 'mosque,' """)


burialTrans =
    transform
        << filter (fiExpr """datum.featureType == 'tomb' ||
                             datum.featureType == 'cemetery'""")


transTwo =
    transform
        << filter (fiExpr """datum.featureType == 'church' ||
                             datum.featureType == 'temple' ||
                             datum.featureType == 'tomb' ||
                             datum.featureType == 'cemetery' ||
                             datum.featureType == 'mosque' ||
                             datum.featureType == 'abbey'
                                 """)


mapData =
    dataFromUrl "https://vega.github.io/vega/data/world-110m.json"
        [ topojsonFeature "countries" ]


mapSpec : Spec
mapSpec =
    asSpec
        [ mapData, geoshape [ maFill "#BBBBBB" ] ]


mapProj =
    projection [ prScale 500, prCenter 50 45, prType naturalEarth1 ]


map : Spec
map =
    let
        churchSpec =
            asSpec
                [ locs
                , settlementEnc []
                , churchTrans []
                , point [ maFilled True, maSize 20, maOpacity 0.6 ]
                ]

        templeSpec =
            asSpec
                [ locs
                , settlementEnc []
                , templeTrans []
                , point [ maFilled True, maSize 20, maOpacity 0.7 ]
                ]

        mosqueSpec =
```

```
        asSpec

            [ locs
            , settlementEnc []
            , mosqueTrans []
            , point [ maFilled True, maSize 20, maOpacity 1 ]
            ]

    burialSpec =
        asSpec
            [ locs
            , settlementEnc []
            , burialTrans []
            , point [ maFilled True, maSize 20, maOpacity 1 ]
            ]
in
toVegaLite [ width 900, height 700, mapProj, layer [ mapSpec, mosqueSpec, burialSpec, templeSpec, churchSpec ] ]
```



What is the name for the type of visualization(s) used?

Map

What are all visual mappings used?

x position

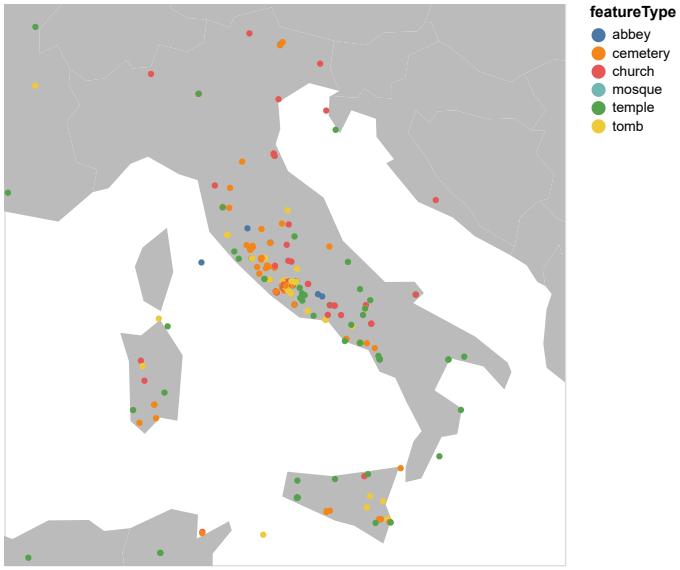
latitude of location

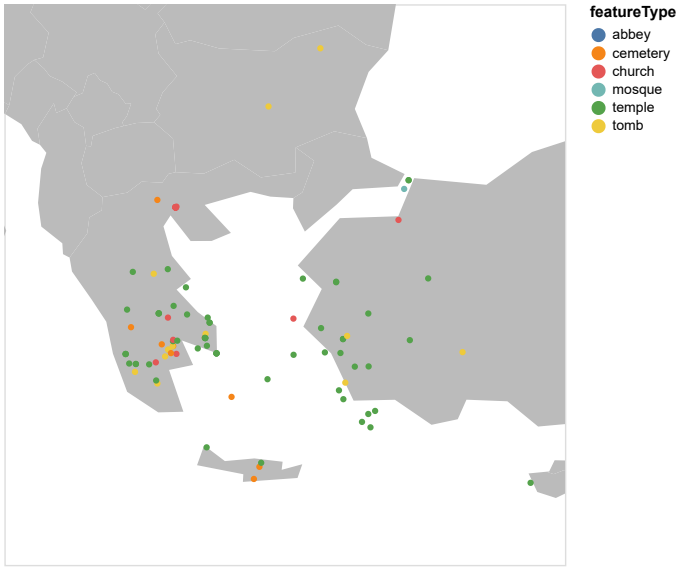
y position

longitude of location

Points are religous sites, they are colour coded by featureType also.

```
italyProj =
    projection [ prScale 1550, prCenter 12.4964 41.9028 ]


mapItaly : Spec
mapItaly =
    let
        italySpec =
            asSpec
                [ locs
                , settlementEnc []
                , transTwo []
                , point [ maFilled True, maSize 20, maOpacity 1 ]
                ]
    in
    toVegaLite [ width 400, height 400, italyProj, layer [ mapSpec, italySpec ] ]


greeceProj =
    projection [ prScale 1550, prCenter 25.8243 39.0742 ]


mapGreece : Spec
mapGreece =
    let
        italySpec =
            asSpec
                [ locs
                , settlementEnc []
                , transTwo []
                , point [ maFilled True, maSize 20, maOpacity 1 ]
                ]
    in
    toVegaLite [ width 400, height 400, greeceProj, layer [ mapSpec, italySpec ] ]
```





Was there any special data preparation done?

Filtering out to only get the `featureTypes` of religous sites.
Aggregating and counting the amounts of each feature
Changing opacity to increase readability and size of some points
Multiple views for dense parts, greece and rome.

Removal of some trailing punctuation to include instances that we not matched by filters

What are the limitations of your design?

The manual zooms provide the features I would like fomr a zoom feature but I struggled to implement this in elm vegalite.
I would have liked to filter by date aswell as I think the rise of churches and drop off of newly constructed temples would be interesting.

Distribution of Latin Name use over time

What can we learn from the visualization?

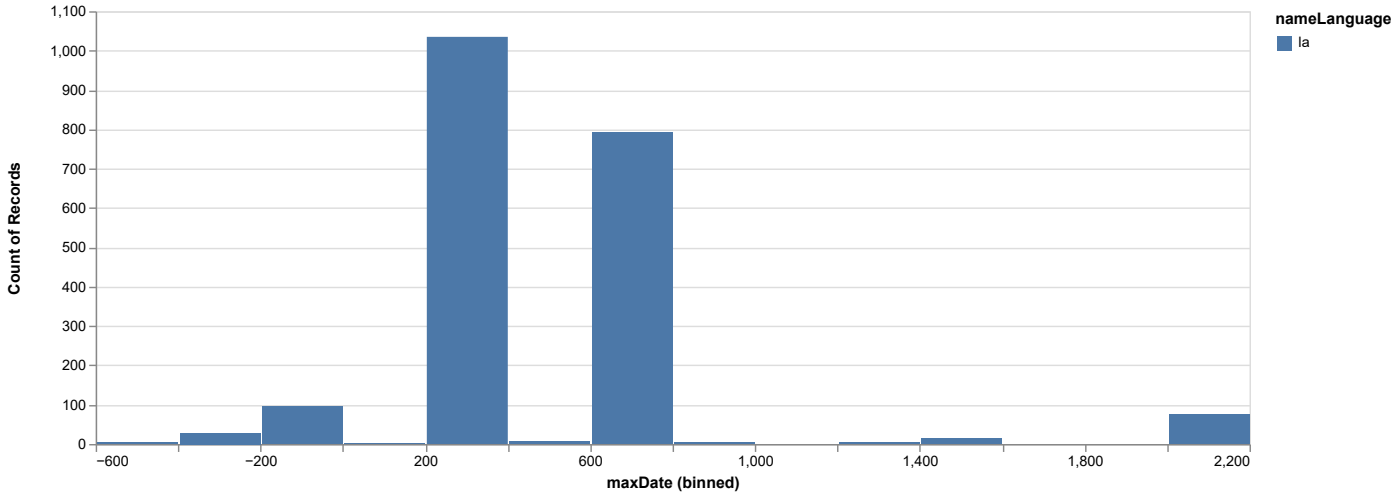
To see how the latin language was being produced over the years.

Many call latin a dying language as it is no longer use in modern culture. The data here suggests that afte the year 800 it is rarely used. We see a slight resurgence after the year 2000 which is strange.

What is the name for the type of visualization(s) used?

Bar Chart

```
barHist : Spec
barHist =
    let
        trans =
            transform
                << filter (fiExpr """datum.minDate >= -1000  &&
                                     (datum.nameLanguage == 'la' )
                                                              """)

        enc =
            encoding
                << position X [ pName "maxDate", pBin [ biStep 200 ] ]
                << position Y [ pName "nameLanguage", pAggregate opCount ]
                << color [ mName "nameLanguage" ]
    in
    toVegaLite [ height 300, width 800, names, trans [], enc [], bar [] ]
```



What are all visual mappings used?

X

maxDate produced with a bin size of 200

y

nameLanguage aggregated and counted together

Was there any special data preparation done?

filtered out when beofre any latin records were given

and set the nameLanguage to 'la' only

What are the limitations of your design?

This is quite a basic graph, I would like to have spent more time on it but was unable to.

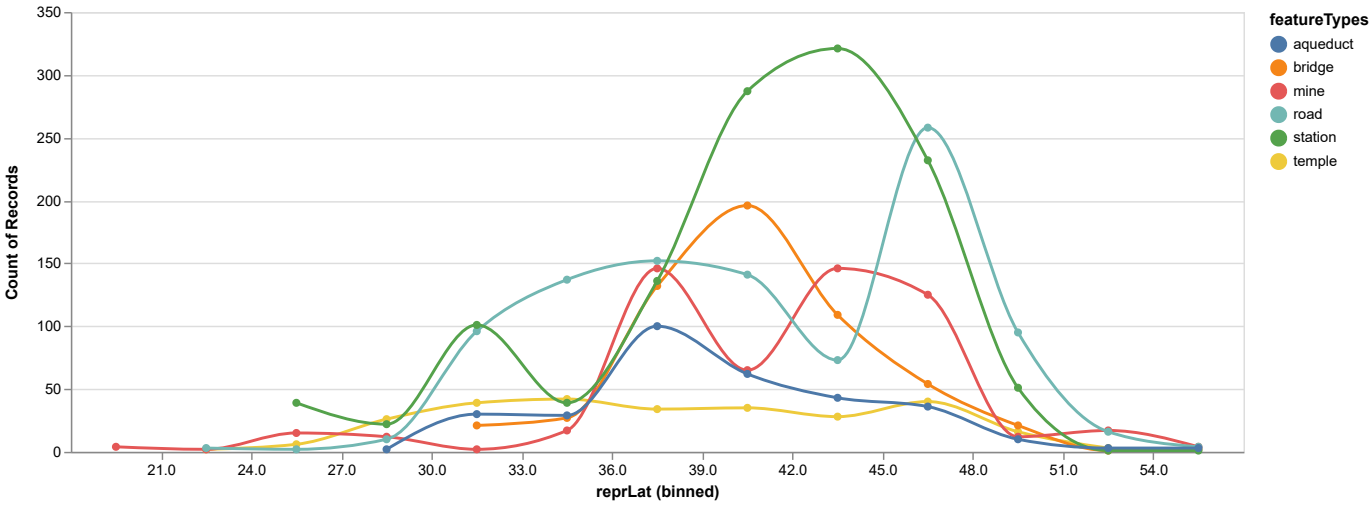
Structure Type by Latitude

What can we learn from the visualization?

This visualisation shows how these chosen technological advances correlate to a certain latitude.

We can see the pattern that the most advanced societies (going by the features I have chosen) are generally from Latitude 36 - 48 degrees. Correlating to the very south of greece and Northern Africa to France and Germany. This is likely due to the Empires being located around these latitudes but also the favourable conditions at this far North.

```
densityDist : Spec
densityDist =
    let
        trans =
            transform
                << filter (fiExpr " datum.reprLat > 20  && datum.reprLat < 61")

        transTemple =
            transform
                << filter (fiExpr """ datum.reprLat != '' &&
                                      datum.featureTypes == 'temple'""")

        transBridge =
            transform
                << filter (fiExpr """ datum.reprLat != '' &&
                                      datum.featureTypes == 'bridge'""")

        transMine =
            transform
                << filter (fiExpr """ datum.reprLat != '' &&
                                      datum.featureTypes == 'mine'""")

        transRoad =
            transform
                << filter (fiExpr """ datum.reprLat != '' &&
                                      datum.featureTypes == 'road'""")

        transStation =
            transform
                << filter (fiExpr """ datum.reprLat != '' &&
                                      datum.featureTypes == 'station'""")

        transAqua =
            transform
                << filter (fiExpr """ datum.reprLat != '' &&
                                      datum.featureTypes == 'aqueduct'""")

        encLat =
            encoding
                << position X
                    [ pName "reprLat"
                    , pBin [ biStep 3 ]
                    ]
                << position Y
                    [ pName "featureTypes"
                    , pAggregate opCount
                    ]
                << color [ mName "featureTypes" ]

        specLine =
            asSpec [ encLat [], transTemple [], line [ maInterpolate miMonotone, maPoint (pmMarker []) ] ]

        specLine1 =
            asSpec [ encLat [], transBridge [], line [ maInterpolate miMonotone, maPoint (pmMarker []) ] ]

        specLine2 =
            asSpec [ encLat [], transMine [], line [ maInterpolate miMonotone, maPoint (pmMarker []) ] ]

        specLine3 =
            asSpec [ encLat [], transRoad [], line [ maInterpolate miMonotone, maPoint (pmMarker []) ] ]

        specLine4 =
            asSpec [ encLat [], transStation [], line [ maInterpolate miMonotone, maPoint (pmMarker []) ] ]

        specLine5 =
            asSpec [ encLat [], transAqua [], line [ maInterpolate miMonotone, maPoint (pmMarker []) ] ]
    in
    toVegaLite [ height 300, width 800, places, trans [], layer [ specLine, specLine1, specLine2, specLine3, specLine4, specLine5
```



What is the name for the type of visualization(s) used?

Line Graph

What are all visual mappings used?

X - Latitudes

Y - Count of `featureTypes`

Was there any special data preparation done?

Filtering of featureTypes

Latitudes have been binned with a value of 3 per bin

Latitude has been filtered from 20 to 60 as there are only some values and the graph is odd proportionally when these are included.

What are the limitations of your design?

Again a checkbox feature may be better to only filter the locations you want to see.

The dataset has a large amount of unknown and blank featureTypes and also latitude columns so this could be skewing the data.