

PROCESS CONTROL CHART



Gian Marco Falcone - 0300251

Indice

- Introduzione: Cosa è un process control chart?
- Introduzione: Scopo del deliverable
- Jira REST API
- Implementazione
- Implementazione: Upper e Lower Limit
- Risultati
- Discussione
- Links

Introduzione: Cosa è un process control chart?

Il process control chart è uno strumento statistico che permette di monitorare il valore di un attributo di un certo processo nel tempo.

Esso permette di:

- Determinare se il processo è stabile rispetto all'attributo di interesse, cioè se il valore che l'attributo assume nel tempo è contenuto in un certo intervallo, oppure se è instabile;
- Predire i valori che l'attributo potrà assumere in futuro, in modo da agire preventivamente.

Introduzione: Scopo del deliverable

Si vuole realizzare un process control chart del progetto software FALCON. Sull'asse delle ordinate va posto l'attributo di interesse sul quale misurare la stabilità: il numero di fixed ticket.

L'asse delle ascisse deve invece necessariamente essere temporalmente ordinato: si è scelto di usare i mesi.

Per raccogliere le informazioni necessarie si utilizza Jira.

Jira REST API

La query utilizzata per ottenere le informazioni necessarie attraverso la Jira REST API è:

- *project = FALCON AND*
- *status = closed OR resolved*
- *AND resolution = fixed*
- *&fields=key,resolutiondate,versions,created*

Volendo ottenere tutte le tipologie di ticket, nessun valore particolare è stato impostato per l'attributo *issuedType*.

Jira REST API

Di seguito un esempio dei risultati ottenuti, da cui sono state ricavate le informazioni necessarie per costruire il process control chart.

Il campo di maggior interesse per l'analisi in questione è *resolutiondate*, corrispondente alla data di chiusura del ticket su Jira.

```
"id": "13152947",  
"self": "https://issues.apache.org/jira/rest/api/2/issue/13152947",  
"key": "FALCON-2336",  
"fields": {  
  "versions": [],  
  "resolutiondate": "2018-08-09T10:25:18.000+0000",  
  "created": "2018-04-17T09:43:20.000+0000"  
}
```

Implementazione

Per ogni ticket restituito da Jira si ottiene il valore di *resolutiondate* e lo si inserisce in una lista di stringhe ordinata temporalmente.

A partire da questa si contano il numero di fixed ticket che presentano lo stesso mese come data di chiusura del ticket, supponendo che ad essa corrisponda anche l'ultimo commit per quel ticket su GitHub.

Questo ha permesso di ottenere un dataset contenente il numero di fixed ticket per ogni mese, i nostri punti sul grafico.

Implementazione: Upper e Lower Limit

Dal dataset creato sono state calcolate anche media e deviazione standard, necessarie per lo sviluppo dei due limiti.

- Upper limit: $\text{media} + 3 \times \text{deviazione standard}$
- Lower limit: $\text{media} - 3 \times \text{deviazione standard}$

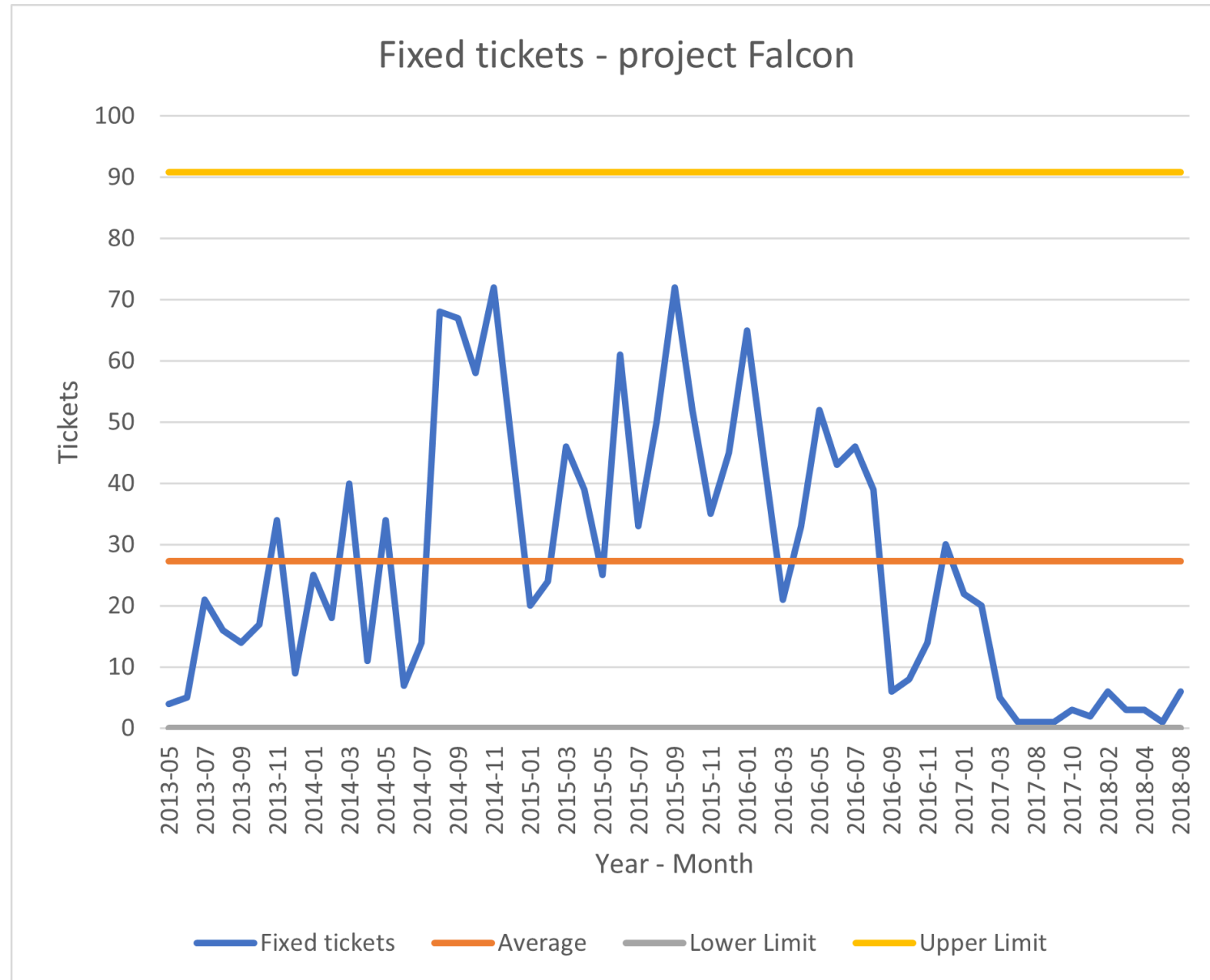
I punti che cadono nell'intervallo tra i due limiti rappresentano quegli stati in cui il processo sotto osservazione mantiene un comportamento stabile.

Al contrario, i punti che cadono al di fuori corrispondono a situazioni significativamente diverse dalle altre, segno di instabilità che può essere dovuta a diversi fattori.

Dovranno quindi essere prese le giuste misure compensative per far ritornare il processo sotto osservazione in una condizione di stabilità.

Media, upper limit e lower limit sono stati aggiunti nel grafico come rette orizzontali.

Risultati



Discussione

In totale sono stati individuati, per il progetto Falcon, 1555 fixed ticket.

Il periodo temporale sotto osservazione va da Maggio 2013, a cui risale il primo fixed ticket su Jira, ad Aprile 2018, data dell'ultimo. Si suppone che da allora il progetto non sia più sotto sviluppo.

Considerando che il lower limit, secondo il dataset ottenuto, avrebbe assunto un valore negativo, e poiché tale valore abbia poco senso per l'attributo sotto osservazione, si è posto il lower limit in corrispondenza dello 0.

Discussione

Si nota che il processo in questione risulta essere stabile, poiché tutti i punti del grafico appartengono all'intervallo, sinonimo che nessun periodo temporale ha avuto una variazione significativa dei difetti.

Ciò nonostante si nota un incremento dei fixed ticket nel periodo tra Novembre 2014 e Settembre 2016, probabilmente a causa di un particolare interesse nello sviluppo del software in quel periodo.

A partire invece da Marzo 2017 il numero di fixed ticket è quasi nullo, segno di un imminente abbandono nello sviluppo del progetto.

Links

Github

- <https://github.com/ShockGiammy/Deliverable1>

SonarCloud

- https://sonarcloud.io/dashboard?id=ShockGiammy_Deliverable1