RFM Analizi

Aşağıdaki e_commerce_data_.csv doyasındaki veri setini kullanarak RFM analizi yapınız. Recency hesaplarken bugünün tarihi değil en son sipariş tarihini baz alınız.

Veri seti bu linkten alınmıştır, veriyi tanımak için linke girip inceleyebilirsiniz.

E-Commerce Data

--En son sipariş tarihi : '2011-12-09 12:50:00'

```
WITH retail AS (
  SELECT
    invoiceno,
    stockcode,
    description,
    quantity,
    invoicedate,
    unitprice,
    customerid,
    country
  FROM retail
  WHERE customerid IS NOT NULL
    AND unitprice > 0
    AND quantity > 0
),
max_invoice_date AS (
  SELECT
    customerid,
    MAX(invoicedate) AS max_i_d
  FROM retail
  GROUP BY customerid
),
recency_table AS (
  SELECT
    customerid,
    max_i_d::DATE,
    '2011-12-09' - DATE(max_i_d) AS recency
  FROM max_invoice_date
),
frequency_table AS (
  SELECT
    customerid,
    COUNT(DISTINCT invoiceno) AS frequency
  FROM retail
```

```
  GROUP BY customerid
),
monetary_table AS (
  SELECT
    customerid,
    SUM(quantity * unitprice) AS monetary
  FROM retail
  GROUP BY customerid
),
scores AS (
  SELECT
    r.customerid,
    r.recency,
    ntile(5) OVER (ORDER BY recency DESC) AS recency_score,
    f.frequency,
    ntile(5) OVER (ORDER BY frequency ASC) AS frequency_score,
    m.monetary,
    ntile(5) OVER (ORDER BY monetary ASC) AS monetary_score
  FROM recency_table r
  LEFT JOIN frequency_table f ON r.customerid = f.customerid
  LEFT JOIN monetary_table m ON f.customerid = m.customerid
),
merge_mon_fre AS (
  SELECT
    customerid,
    recency_score,
    frequency_score + monetary_score AS mon_fre_score
  FROM scores
),
rfm_scores AS (
  SELECT
    customerid,
    recency_score,
    ntile(5) OVER (ORDER BY mon_fre_score ASC) AS monetary_frequency_merged_score
  FROM merge_mon_fre
),
rfm_scores_with_segments AS (
  SELECT
    customerid,
    recency_score,
    monetary_frequency_merged_score,
    recency_score || '' || monetary_frequency_merged_score AS rfm_score,
    CASE
      WHEN CONCAT(recency_score::TEXT, monetary_frequency_merged_score::TEXT) IN ('11', '22', '12', '21')
THEN '10_hibernating'
      WHEN CONCAT(recency_score::TEXT, monetary_frequency_merged_score::TEXT) IN ('14', '24', '13', '23')
THEN '09_at_Risk'
      WHEN CONCAT(recency_score::TEXT, monetary_frequency_merged_score::TEXT) IN ('15', '25') THEN
'08_cant_loose'
```

```
        WHEN CONCAT(recency_score::TEXT, monetary_frequency_merged_score::TEXT) IN ('33') THEN
'07_about_to_sleep'
        WHEN CONCAT(recency_score::TEXT, monetary_frequency_merged_score::TEXT) IN ('31', '32') THEN
'06_need_attention'
        WHEN CONCAT(recency_score::TEXT, monetary_frequency_merged_score::TEXT) IN ('34', '35', '44', '45')
THEN '02_loyal_customers'
        WHEN CONCAT(recency_score::TEXT, monetary_frequency_merged_score::TEXT) = '41' THEN
'05_promising'
        WHEN CONCAT(recency_score::TEXT, monetary_frequency_merged_score::TEXT) = '51' THEN
'04_new_customers'
        WHEN CONCAT(recency_score::TEXT, monetary_frequency_merged_score::TEXT) IN ('42', '43', '52', '53')
THEN '03_potential_loyalists'
        WHEN CONCAT(recency_score::TEXT, monetary_frequency_merged_score::TEXT) IN ('54', '55') THEN
'01_champions'
        ELSE 'unknown'
      END AS rfm_segment
   FROM rfm_scores
),
rfm_distribution AS (
   SELECT rfm_segment,
      COUNT(rfm_segment) AS customer_count
   FROM rfm_scores_with_segments
   GROUP BY 1
),
rfm_percentages AS (
   SELECT
      rfm_segment,
      customer_count,
      ROUND(customer_count::NUMERIC / SUM(customer_count) OVER (), 2) AS segment_percentage
   FROM rfm_distribution
),
rfm_recommendations AS (
   SELECT
      rfm_segment,
      customer_count,
      ROUND(customer_count::NUMERIC / SUM(customer_count) OVER (), 2) AS segment_percentage,
      CASE
        WHEN rfm_segment = '01_champions' THEN 'Yeni ve yaklaşan ürünleri tanıtın ve duyurun. Onları
ödüllendirin ve güncellemeleri paylaşmalarına yardımcı olun.'
        WHEN rfm_segment = '02_loyal_customers' THEN 'Daha yüksek değerli ürünleri satmaya çalışın.
Değerlendirmelerini isteyin ve onlarla etkileşimde bulunun.'
        WHEN rfm_segment = '03_potential_loyalists' THEN 'Üyelik veya sadakat programı teklif edin ve diğer
ürünleri tavsiye edin.'
        WHEN rfm_segment = '04_new_customers' THEN 'Başlangıç desteği sağlayın, erken başarılarını teşvik
edin ve ilişki kurmaya başlayın.'
        WHEN rfm_segment = '05_promising' THEN 'Yeniden tüketim ihtiyaçlarını kontrol edin, geri bildirim
isteyin ve en popüler ürünleri paylaşın.'
        WHEN rfm_segment = '06_need_attention' THEN 'Sınırlı süreli teklifler sunun, geçmiş alışverişlere dayalı
önerilerde bulunun ve bu müşterileri yeniden etkinleştirin.'
```

```
        WHEN rfm_segment = '07_about_to_sleep' THEN 'Ilgilerini cekebilecek icerikler sunun. Indirimli fiyatlarla popüler ürünleri önerin ve bu müşterilerle yeniden iletişim kurun.'
        WHEN rfm_segment = '08_cant_loose' THEN 'Markanızı ilk neden sevdiklerini hatırlatın ve ilişkiyi canlandırın.'
        WHEN rfm_segment = '09_at_Risk' THEN 'Kişiselleştirilmiş e-postalar göndererek yeniden bağlantı kurun, yenileme teklif edin ve yardımcı kaynaklar sağlayın.'
        WHEN rfm_segment = '10_hibernating' THEN 'İlgili diğer ürünler ve özel indirimler sunun, marka değerini yeniden oluşturun.'
      END AS recommendations
   FROM rfm_percentages
)
SELECT *
FROM rfm_recommendations
ORDER BY 1;
```


```
-- seg_map = {
--     r'[1-2][1-2]': 'hibernating',
--     r'[1-2][3-4]': 'at_Risk',
--     r'[1-2]5': 'cant_loose',
--     r'3[1-2]': 'about_to_sleep',
--     r'33': 'need_attention',
--     r'[3-4][4-5]': 'loyal_customers',
--     r'41': 'promising',
--     r'51': 'new_customers',
--     r'[4-5][2-3]': 'potential_loyalists',
--     r'5[4-5]': 'champions'
-- }
```

- Paylaşılan veri setinde RFM Analizi yapılmış, herbir müşteri için recency, frequency ve monetary değerleri hesaplanmış, skorlara dönüştürülmüş, F ve M skorları birleştirilerek yeni bir skor oluşturulmuş ve bu yeni skor, R değeri ile birlikte düşünülerek yukarıdaki skor haritasına göre tüm müşteriler segmentlere ayrılmıştır.
- Her segment içerisinde bulunan müşteri sayıları ve her segment içerisinde bulunan müşteri grubu için alınması gerekli aksiyonlar sql sorgusunun içerisinde verilmiştir.
- Segmentlere göre yapılmış öneriler aşağıda ayrıca verilmiştir:

Champions : *Yeni ve yaklaşan ürünleri tanıtın ve duyurun. Onları ödüllendirin ve güncellemeleri paylaşmalarına yardımcı olun.*

Loyal Customers : *Daha yüksek değerli ürünleri satmaya çalışın. Değerlendirmelerini isteyin ve onlarla etkileşimde bulunun.*

Potential Loyalists : *Üyelik veya sadakat programı teklif edin ve diğer ürünleri tavsiye edin.*

New Customers : *Başlangıç desteği sağlayın, erken başarılarını teşvik edin ve ilişki kurmaya başlayın.*

Promising : *Yeniden tüketim ihtiyaçlarını kontrol edin, geri bildirim isteyin ve en popüler ürünleri paylaşın.*

Need Attention : *Sınırlı süreli teklifler sunun, geçmiş alışverişlere dayalı önerilerde bulunun ve bu müşterileri yeniden etkinleştirin.*

About to Sleep : *Ilgilerini cekebilecek icerikler sunun. Indirimli fiyatlarla popüler ürünleri önerin ve bu müşterilerle yeniden iletişim kurun.*

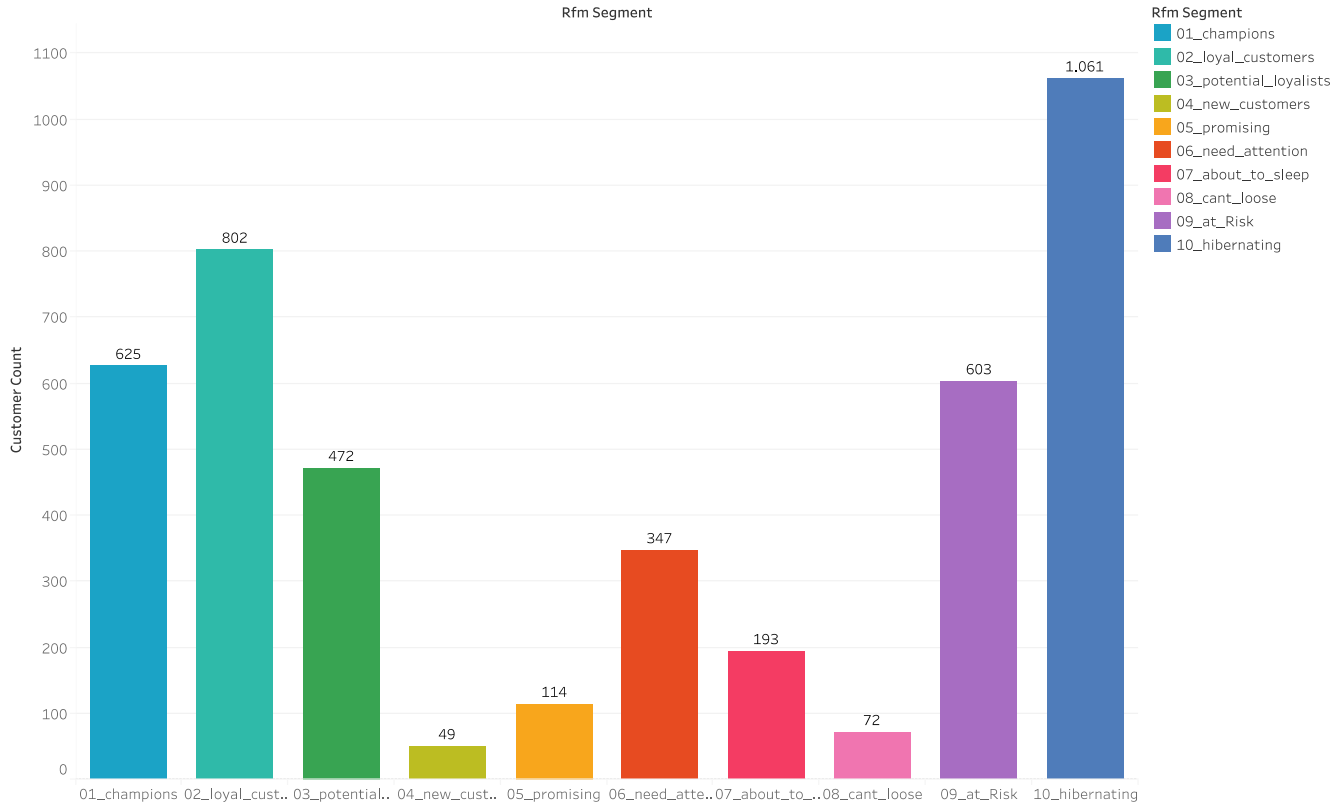
Can't Loose : *Markanızı ilk neden sevdiklerini hatırlatın ve ilişkiyi canlandırın.*

At Risk : *Kişiselleştirilmiş e-postalar göndererek yeniden bağlantı kurun, yenileme teklif edin ve yardımcı kaynaklar sağlayın.*

Hibernating : *İlgili diğer ürünler ve özel indirimler sunun, marka değerini yeniden oluşturun.*

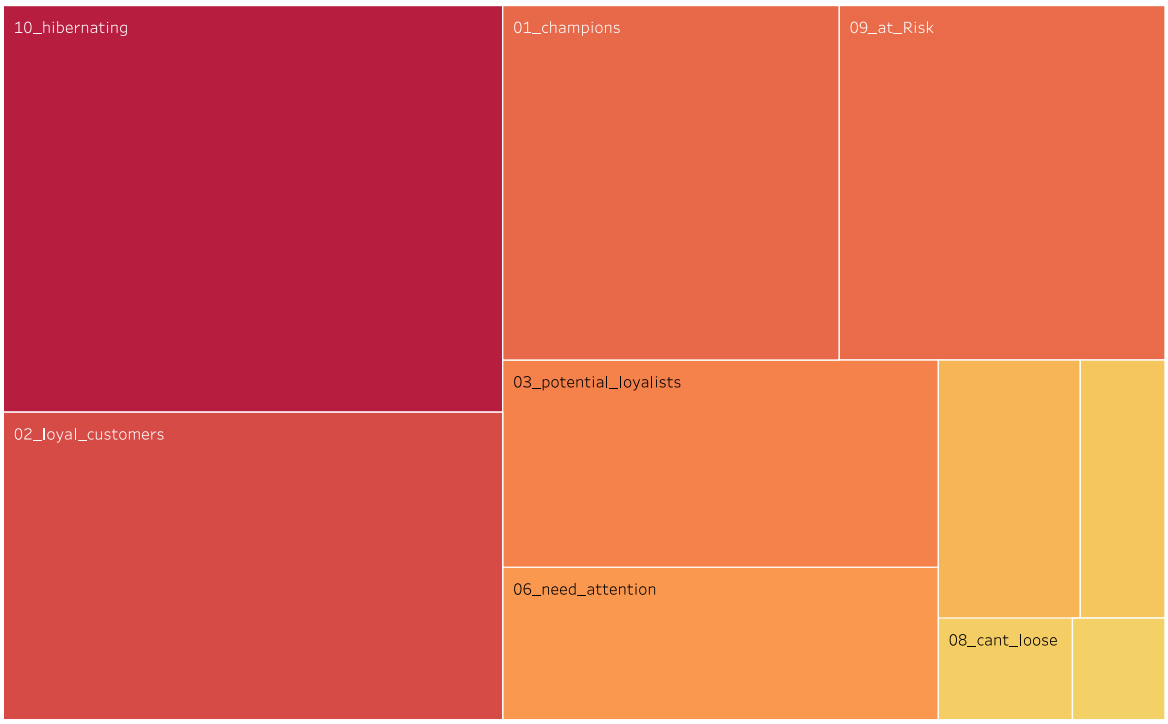
- SQL sorgusu ile yapılmış olan analiz sonucunda oluşan çıktının sonucunda her segment için bulunan müşteri sayıları aşağıda verilmiştir:

RFM Segments - Count

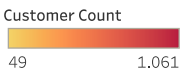


Sum of Customer Count for each Rfm Segment. Color shows details about Rfm Segment. The marks are labeled by sum of Customer Count.

RFM Segments - Count



Rfm Segment. Color shows sum of Customer Count. Size shows sum of Customer Count. The marks are labeled by Rfm Segment.



- Ayrıca, her segment için ortalama satın alma sıklığını, ortalama parasal değeri ve ortalama recency değerini hesapladım:

```
WITH retail AS (
  SELECT
    invoiceno,
    stockcode,
    description,
    quantity,
    invoicedate,
    unitprice,
    customerid,
    country
  FROM retail
  WHERE customerid IS NOT NULL
    AND unitprice > 0
    AND quantity > 0
),
max_invoice_date AS (
  SELECT
    customerid,
    MAX(invoicedate) AS max_i_d
  FROM retail
  GROUP BY customerid
),
recency_table AS (
  SELECT
    customerid,
    max_i_d::DATE,
    '2011-12-09' - DATE(max_i_d) AS recency
  FROM max_invoice_date
),
frequency_table AS (
  SELECT
    customerid,
    COUNT(DISTINCT invoiceno) AS frequency
  FROM retail
  GROUP BY customerid
),
monetary_table AS (
  SELECT
    customerid,
    SUM(quantity * unitprice) AS monetary
  FROM retail
  GROUP BY customerid
),
scores AS (
  SELECT
```

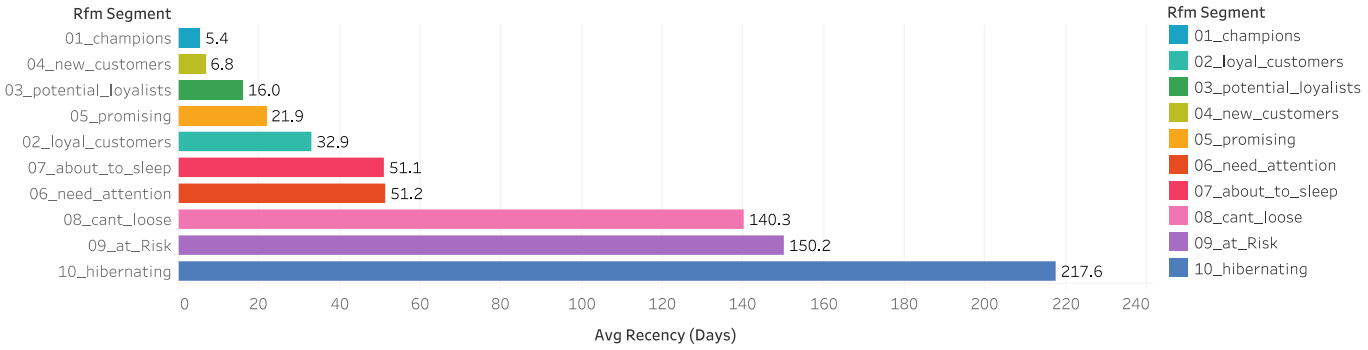
```
    r.customerid,
    r.recency,
    NTILE(5) OVER(ORDER BY recency DESC) AS recency_score,
    f.frequency,
    NTILE(5) OVER(ORDER BY frequency ASC) AS frequency_score,
    m.monetary,
    NTILE(5) OVER(ORDER BY monetary ASC) AS monetary_score
  FROM recency_table r
  LEFT JOIN frequency_table f ON r.customerid = f.customerid
  LEFT JOIN monetary_table m ON f.customerid = m.customerid
),
merge_mon_fre AS (
  SELECT
    customerid,
    recency_score,
    frequency_score + monetary_score AS mon_fre_score
  FROM scores
),
rfm_scores AS (
  SELECT
    customerid,
    recency_score,
    NTILE(5) OVER(ORDER BY mon_fre_score ASC) AS monetary_frequency_merged_score
  FROM merge_mon_fre
),
rfm_scores_with_segments AS (
  SELECT
    customerid,
    recency_score,
    monetary_frequency_merged_score,
    recency_score || '' || monetary_frequency_merged_score AS rfm_score,
    CASE
      -- "seg_map" kullanarak RFM segmentlerini eşleştiriyoruz
      WHEN CONCAT(recency_score::TEXT, monetary_frequency_merged_score::TEXT) IN ('11', '22', '12', '21')
THEN '10_hibernating'
      WHEN CONCAT(recency_score::TEXT, monetary_frequency_merged_score::TEXT) IN ('14', '24', '13', '23')
THEN '09_at_Risk'
      WHEN CONCAT(recency_score::TEXT, monetary_frequency_merged_score::TEXT) IN ('15', '25') THEN
'08_cant_loose'
      WHEN CONCAT(recency_score::TEXT, monetary_frequency_merged_score::TEXT) IN ('33') THEN
'07_about_to_sleep'
      WHEN CONCAT(recency_score::TEXT, monetary_frequency_merged_score::TEXT) IN ('31', '32') THEN
'06_need_attention'
      WHEN CONCAT(recency_score::TEXT, monetary_frequency_merged_score::TEXT) IN ('34', '35', '44', '45')
THEN '02_loyal_customers'
      WHEN CONCAT(recency_score::TEXT, monetary_frequency_merged_score::TEXT) = '41' THEN
'05_promising'
      WHEN CONCAT(recency_score::TEXT, monetary_frequency_merged_score::TEXT) = '51' THEN
'04_new_customers'
```

```
        WHEN CONCAT(recency_score::TEXT, monetary_frequency_merged_score::TEXT) IN ('42', '43', '52', '53')
THEN '03_potential_loyalists'
        WHEN CONCAT(recency_score::TEXT, monetary_frequency_merged_score::TEXT) IN ('54', '55') THEN
'01_champions'
        ELSE 'unknown'
      END AS rfm_segment
   FROM rfm_scores
),
rfm_scores_with_segments_1 AS (
   SELECT
      customerid,
      recency_score,
      monetary_frequency_merged_score,
      rfm_segment
   FROM rfm_scores_with_segments
),
segment_metrics AS (
   SELECT
      rfm_segment,
      AVG(scores.frequency) AS avg_frequency,
      AVG(scores.monetary) AS avg_monetary,
      AVG(scores.recency) AS avg_recency
   FROM rfm_scores_with_segments_1
   LEFT JOIN scores ON rfm_scores_with_segments_1.customerid = scores.customerid
   GROUP BY rfm_segment
)

SELECT
   rfm_segment,
   avg_frequency,
   avg_monetary,
   avg_recency
FROM segment_metrics
ORDER BY rfm_segment;
```

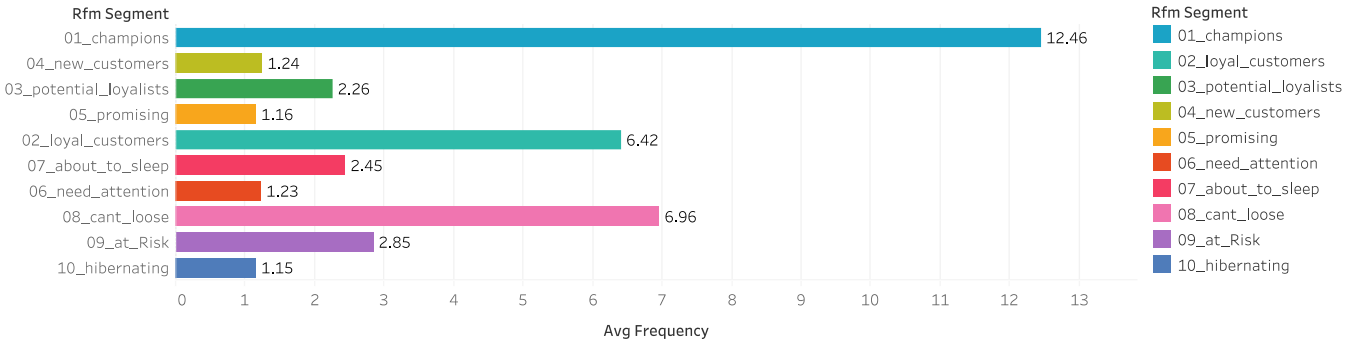
Her segment için hesaplanmış ortalama recency, frequency ve monetary değerleri aşağıdaki grafik ile görselleştirilmiştir:

Average Recency By Segments



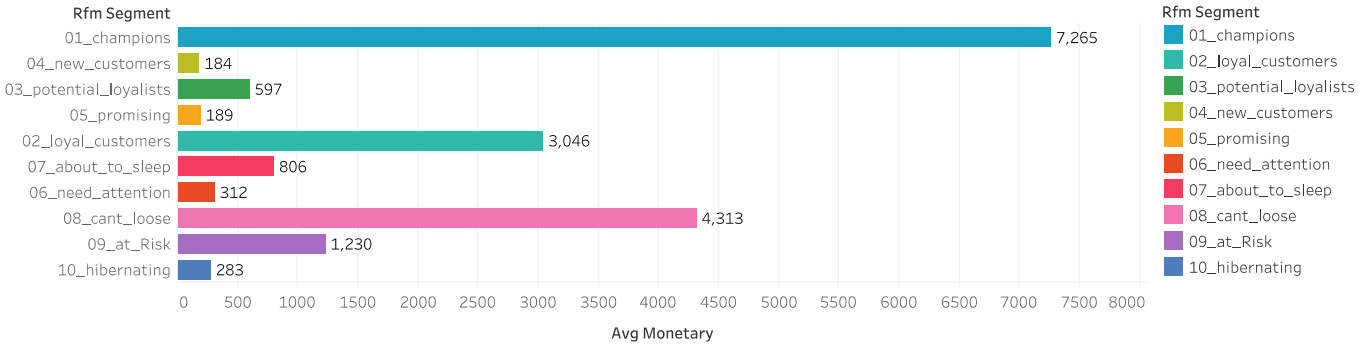
Sum of Avg Recency for each Rfm Segment. Color shows details about Rfm Segment. The marks are labeled by sum of Avg Recency.

Average Frequency By Segments



Sum of Avg Frequency for each Rfm Segment. Color shows details about Rfm Segment. The marks are labeled by sum of Avg Frequency.

Average Monetary By Segments



Sum of Avg Monetary for each Rfm Segment. Color shows details about Rfm Segment. The marks are labeled by sum of Avg Monetary.